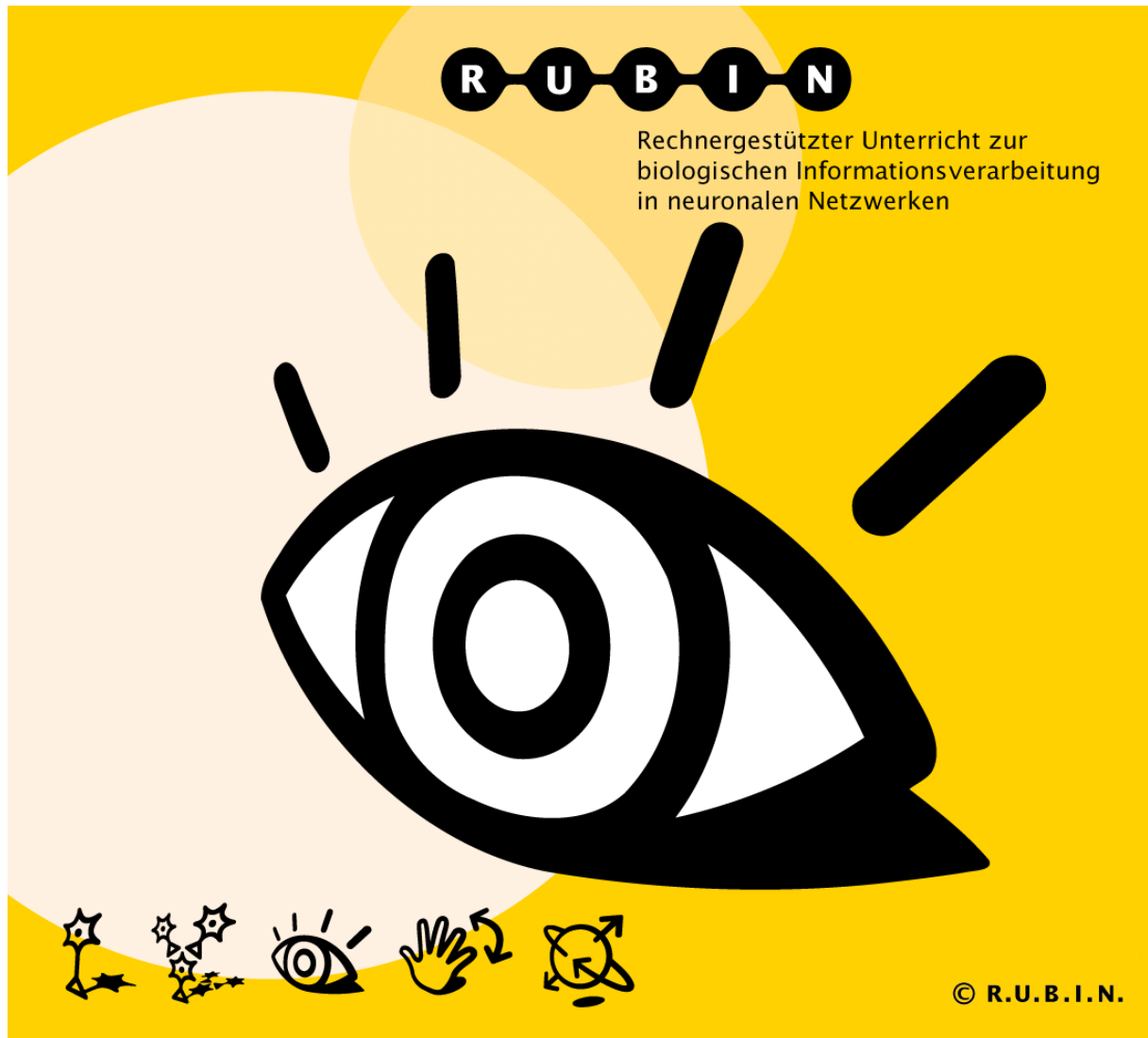


# Projektbeschreibung



Stand: 1.11.1999



# Inhaltsverzeichnis

A.	<b>Kurzbeschreibung</b>	5
1.	Übersicht	5
2.	Motivation	5
3.	Entwicklungsziel, Zielgruppe und Zielplattform	5
4.	Module	6
5.	Umsetzung	6
B.	<b>Motivation</b>	7
1.	Eine Lücke zwischen Forschung und Lehre	7
2.	Mediendidaktische Ausnahmestellung	7
3.	Wissenschaftliches und öffentliches Interesse	8
4.	Auslastung bestehender Infrastruktur und Technik	8
C.	<b>Konzept</b>	9
1.	Lehrsimulation	9
1.1.	Einführung	9
1.2.	Interaktive Visualisierung	10
1.2.1.	Simulationsumgebung	10
1.2.2.	Oszilloskop	12
1.2.3.	Plotter	13
1.2.4.	Pixeldarstellung und Falschfarbencodierung	13
1.2.5.	Interaktive Animationen	14
1.2.6.	Virtuelle Räume und 3D-Animation	15
1.2.7.	Formeln	16
1.3.	Fließtext	16
1.4.	Unterstützende Medien	16
1.5.	Lernzielkontrolle	16
2.	Simulationslehrsystem	17
2.1.	Motivation	17
2.2.	Gliederung	17
2.3.	Inhalte	18
2.4.	Simulationstechnik	21
2.5.	Weitere Eigenschaften des Simulationslehrsystems	22
2.5.1.	Benutzerführung	22
2.5.2.	Multimediales Glossar	22
2.5.3.	Hilfesystem	23

3.	Lernumgebung	24
3.1.	Motivation	24
3.2.	»Virtuelles Tier«	24
3.3.	Benutzerprofil	25
3.4.	Lernwegvisualisierungen	25
3.5.	Lernwegauswahl	25
3.6.	Lern- und Lehrplaner: Evaluation	26
4.	Entwicklungsumgebung	26
4.1.	Motivation	26
4.2.	Simulationswerkzeug	27
4.3.	Skripter	27
5.	Online-Umgebung	28
5.1.	Motivation	28
5.2.	RUBIN-Kommunikationsschnittstelle	28
5.3.	Kooperatives Lernen	28
5.4.	WBT-Erweiterung für den Lern- und Lehrplaner	28
5.5.	Server	29
5.5.1.	Administration	29
5.5.2.	Datenbank	29
D.	<b>Realisierung</b>	30
1.	Basisversion: Simulationslehrsystem	30
1.1.	Vorarbeiten	30
1.2.	Produktionsaufwand	31
1.2.1.	Autorentätigkeiten	31
1.2.2.	Medienproduktion	31
1.2.3.	Programmierung	31
1.2.4.	Qualitätssicherung	32
1.2.5.	Koordination	32
2.	Ausbaustufen: Lern-, Entwicklungs- und Onlineumgebung	32
2.1.	Vorarbeiten	32
2.2.	Produktionsaufwand	32
2.2.1.	Lernumgebung	32
2.2.2.	Entwicklungsumgebung	33
2.2.3.	Online-Umgebung	33
3.	Kooperationen	34
3.1.	Beteiligte Einrichtungen	34
3.2.	Geplante Kooperationen	35

# RUBIN

Rechnergestützter Unterricht zur  
Biologischen Informationsverarbeitung in neuronalen Netzwerken

~ Projektbeschreibung ~

## A. Kurzbeschreibung

### 1. Übersicht

RUBIN ist ein multimediales und interaktives Lern- und Lehrsystem im Bereich der Neurowissenschaften, in dem Lernende in didaktisch aufbereiteten Lehrsimulationen erfahren können, wie Informationen im Gehirn verarbeitet werden. Um dem stark interdisziplinären Charakter der Thematik gerecht zu werden, ist das Ziel des Projekts, einen gemeinsamen Rahmen für die Lehre in Biologie, Informatik und kognitionswissenschaftlicher Disziplinen vorzuschlagen. Als vollständig in JAVA realisiertes Lern- und Lehrsystem mit einer graphischen Benutzeroberfläche, aus der heraus alle bereitgestellten Module erreichbar sind, ist RUBIN prinzipiell auf jeder Rechner- und Netzwerkarchitektur lauffähig und eignet sich damit für das Selbststudium am Heimrechner, zum netzbasierten Lernen in Gruppen an universitären Servern sowie für punktuelle Präsentationen in mit Multimedialechnik ausgestatteten Veranstaltungsräumen. Die modular angelegte Struktur von RUBIN erlaubt die Anbindung vieler weiterer Funktionseinheiten, die das Lernen und Lehren effektiver und komfortabler machen und RUBIN letztendlich zu einem Lern- und Lehrsystem mit Modellcharakter machen können.

### 2. Motivation

Die Neurowissenschaften haben gerade bei der Aufklärung der Prinzipien von Informationsverarbeitung im Gehirn in den letzten Jahren enorme Fortschritte erzielt, die zu einem großen Teil durch die Möglichkeiten bedingt sind, die zugrundeliegenden Prozesse auf dem Computer zu simulieren. Diese Fortschritte sind der wachsenden Anzahl von Interessierten jedoch noch nicht zugänglich. Die Thematik erfordert eine mediendidaktische Aufarbeitung, die bisher nicht realisiert wurde, obwohl alle technischen und infrastrukturellen Voraussetzungen dafür gegeben sind.

### 3. Entwicklungsziel, Zielgruppe und Zielplattform

Ziel des Projekts RUBIN ist es, Lernenden und Lehrenden didaktische Werkzeuge in einem geschlossenen Lern- und Lehrsystem an die Hand zu geben, mit denen in Vorträgen, in Gruppenräumen oder im Selbststudium Informationsverarbeitung im Gehirn veranschaulicht werden kann.

Die Zielgruppe besteht vornehmlich aus den Lernenden und Lehrenden der beteiligten Disziplinen, also der Biologie, der Informatik und den Kognitionswissenschaften. Dies können Studierende im Grund- und Hauptstudium, aber auch Schüler der gymnasialen Oberstufe sein. Um eine möglichst breite und flexible Einsetzbarkeit zu erreichen und Erweiterbarkeit zu gewährleisten, wird RUBIN vollständig in JAVA realisiert. Damit ist RUBIN prinzipiell auf allen Rechnerarchitekturen lauffähig und dafür prädestiniert, im Netzwerkverbund angewendet zu werden. Ferner eröffnet JAVA die Möglichkeit, eine ansprechende und einheitliche graphische Benutzeroberfläche bereitzustellen, über die komfortabel auf alle Module des Lern- und Lehrsystems zugegriffen werden kann.

## 4. Module

Rubin ist modular organisiert. Das Hauptmodul ist das Simulationslehrsystem, das eine Übersicht über die Modelle und Simulationstechniken der Neurowissenschaften gibt. Das Simulationslehrsystem besteht aus zahlreichen interaktiven Lehrsimulationen mit dynamischer Visualisierung. Die Lehrsimulationen sind in einen linearen Lernweg eingebettet, der vom Neuron über Netzwerke bis zu einfachem Verhalten führt. Die in sich geschlossene Struktur der Lehrsimulationen erlaubt es jedoch auch, sie als einzelstehende Lerneinheiten zu verwenden.

In dieser Basisversion eignet sich RUBIN für das Selbststudium oder für punktuelle Präsentationen in universitären Veranstaltungen. Um RUBIN zu einem komfortablen und effektiven Lern- und Lehrwerkzeug zu erweitern, sind drei Ausbaustufen vorgesehen: eine Lernumgebung, in der Lernende ihren Wissensstand überprüfen, konsolidieren und beurteilen können, eine Entwicklungsumgebung, durch die Lehrende und Lernende befähigt werden, eigene Lehrsimulationen zu erstellen und eine Online-Umgebung, in der Gruppenveranstaltungen organisiert und kooperatives Lernen gefördert werden kann.

## 5. Umsetzung

Das Projekt RUBIN wird seit dem 1.7.1998 vom Universitätsverbund MultiMedia des Landes NRW gefördert. Die Projektlaufzeit endet am 31.12.1999. Die Personalmittel (ca. 20 Mannmonate auf 18 Kalendermonate verteilt) und Sachmittel wurden genutzt, ein umfassendes Konzept zu entwickeln und eine personelle und technische Infrastruktur zu etablieren, in der ein technisch und graphisch ausgereifter Prototyp sowie eine Entwicklungsumgebung für die professionelle und schnelle Produktion von Lerneinheiten und Zusatzmodulen erstellt wurde.

Die Vorarbeiten erlauben – bei entsprechender personeller Ausstattung – die Produktion eines umfangreichen, in sich geschlossenen Lernprogramms (siehe Abb.17 Lehrsimulationen & Simulationslehrsystem) innerhalb eines Kalenderjahres. Ende 2000 könnte damit die Basisversion von RUBIN vertrieben werden.

Die modular und plattformübergreifend angelegte JAVA-Struktur und die mediendidaktische Ausnahmestellung prädestinieren RUBIN über das Jahr 2000 hinaus, in Ausbaustufen weiterentwickelt und durch die Integration der Funktionsbereiche Lernumgebung, Entwicklungsumgebung und Online-Umgebung ein Lern- und Lehrsystem mit Modellcharakter zu werden.

# RUBIN

Rechnergestützter Unterricht zur  
Biologischen Informationsverarbeitung in neuronalen Netzwerken

~ Projektbeschreibung ~

## B. Motivation

### 1. Eine Lücke zwischen Forschung und Lehre

Neurobiologen, Informatikern und Kognitionswissenschaftlern ist es in den letzten Jahrzehnten gelungen, Modelle zu entwickeln, die erklären können, wie Tiere und Menschen Informationen aus ihrer Umwelt aufnehmen und so verarbeiten, daß sie sich in ihr zurechtfinden. Elementar für diese Entwicklung war, daß Dynamik und Komplexität der Informationsverarbeitung in Modellsimulationen auf dem Computer umfassend analysiert, visualisiert und evaluiert werden konnten.

Die wissenschaftliche Forschung hat also gezeigt, daß der Weg zu einem umfassenden Verständnis von Informationsverarbeitung in lebenden Systemen zwangsläufig über Modellsimulationen führt. Trotzdem sind die Techniken der Analyse, Visualisierung und Evaluation in Modellsimulationen den meisten Lernenden, die sich mit der Thematik beschäftigen, bis heute nicht zugänglich gemacht worden. Die derzeit verfügbare Simulations-Software stammt aus der wissenschaftlichen Forschung und erfordert ein großes Maß an Programmiererfahrung und mathematischen Vorkenntnissen, so daß sie nur von einer kleinen Gruppe, etwa Informatikstudierenden im Hauptstudium, bedient werden kann. Daher ist es in weniger formal orientierten Disziplinen, wie der Biologie oder kognitionswissenschaftlicher Disziplinen, lediglich engagierten Autodidakten möglich, über Modellsimulationen ein Gefühl für die Dynamik und Komplexität von Informationsverarbeitung im Gehirn zu entwickeln und letztlich auch in Personalunion wissenschaftlich experimentieren und modellieren zu können. Eine Motivation für das RUBIN-Projekt ist also, Werkzeuge entwickeln zu wollen, die helfen, die zwischen Forschung und Lehre bestehende didaktische Lücke zu schließen.

### 2. Mediendidaktische Ausnahmestellung

In den letzten Jahren sind – gerade im Bereich des ‘Computer-Based-Training’ (CBT) – eine Reihe sogenannter Multimedia-Anwendungen produziert worden, die im Endeffekt nichts anderes machen, als herkömmliche Lehrbücher auf den Computer zu übertragen. Die mediendidaktische Grundlage solcher Anwendungen ist fraglich.

Informationsverarbeitung im Gehirn kann nur mit Hilfe des Computers umfassend verstanden werden. Das Thema besitzt eine mediendidaktische Ausnahmestellung: Wenn Lehrende erklären wollen, wie Informationsverarbeitung im Gehirn funktioniert, stoßen sie unweigerlich auf ein Problem: Informationsverarbeitung ist in erster Linie ein Prozeß. Das Lernen und Vermitteln von Prozessen unterscheidet sich grundsätzlich vom Lernen und Vermitteln von Fakten: Faktenwissen (deklaratives Wissen) läßt sich leicht in Worte fassen, während prozedurales Wissen durch Handlungen und Interaktionen erworben werden kann. Die herkömmlichen Lehrbücher sowie die gängigen universitären Veranstaltungsformen (Vorlesungen und Seminare), mit denen Informationsverarbeitung gelehrt und gelernt wird, beruhen jedoch primär auf verbaler Vermittlung, so daß die prozedurale Komponente der Informationsverarbeitung unberührt bleibt. Experimentelle Praktika könnten zwar prozedurales Wissen vermitteln, bieten jedoch aufgrund der aufwendigen und empfindlichen Meßtechnik lediglich fragmenthafte Einblicke in den Informationsverarbeitungsprozeß. Die Interaktivität und dynamische Visualisierung, die über den Computer realisierbar sind, schaffen hier Abhilfe.

Nicht ohne Grund macht sich die wissenschaftliche Forschung eben diese Möglichkeiten zunutze und simuliert die in der experimentellen Forschung entwickelten Modelle von Informationsverarbeitungsprozessen auf dem Computer.

### 3. **Wissenschaftliches und öffentliches Interesse**

Durch die Entwicklung neuer Methoden und den damit verbundenen wissenschaftlichen Erfolgen, ist in den letzten Jahren ein regelrechter »Neuro-Boom« zu beobachten. Berühmte Zeitgenossen bezeichnen die Neunziger Jahre als das »Decade of the Brain«, Wissenschaftstheoretiker prognostizieren den Neurowissenschaften, die Molekularbiologie als Jahrhundertwissenschaft abzulösen. Die erstaunlichen Fortschritte in der medizinischen Diagnostik und Intervention, vor allem aber die Entwicklung hochauflösender bildgebender Verfahren und rechnergestützter Visualisierungen haben dazu beigetragen, daß die Neurowissenschaften Eingang in die Massenmedien gefunden haben. Zwar ist das Ziel von RUBIN weder, diese Entwicklung der Neurowissenschaften vollständig abzubilden, noch, einen Massenmarkt anzusprechen – das Ziel von RUBIN ist jedoch, bestimmte, für jedermann relevante Fragestellungen der Neurowissenschaft herauszugreifen und didaktisch so aufzubereiten, daß die wissenschaftliche und wissenschaftspropädeutische Ausbildung bis hinab zur gymnasialen Oberstufe davon profitieren kann.

### 4. **Auslastung bestehender Infrastruktur und Technik**

In den letzten Jahren sind große Investitionen in den Ausbau der Infrastruktur für den Netzverkehr getätigt und viele Fortschritte in der Verfügbarkeit und Kommunizierbarkeit von Daten (z.B. Deutsches Forschungsnetz, »Schulen ans Netz«) erzielt worden. Die Lehre hat bisher davon nicht in ähnlicher Weise profitieren können wie z.B. das Bibliothekssystem. Die Lernenden benutzen zwar Rechner in Bibliotheken, Cip-Pools, Rechenzentren oder daheim, um zu recherchieren oder Dokumente zu verfassen. Es ist jedoch nicht der Normalfall, daß sie mit dem Computer Inhalte oder Methoden lernen, indem sie mit einer multimedialen Lernumgebung interagieren oder multimedialen Vorträgen ihrer Lehrenden lauschen. Ob der Grund hierfür an der minderwertigen Qualität oder schlicht am Fehlen von Angeboten liegt, sei dahingestellt. Deutlich wird jedoch, daß die multimediale Lehre den technischen Möglichkeiten hinterherläuft. In RUBIN sollen die vielfältigen Möglichkeiten genutzt und exemplarisch für die Neurowissenschaften ein Lern- und Lehrsystem entwickelt werden, das sowohl Lernende dazu anregt, den Computer zum Lernen von Inhalten und Techniken zu verwenden, als auch Lehrende bewegt, multimediale Präsentationen in Vorträgen zu nutzen, wobei die Kommunikationsmöglichkeiten in Netzwerken sinnvoll eingesetzt werden können.



# RUBIN

## Rechnergestützter Unterricht zur Biologischen Informationsverarbeitung in neuronalen Netzwerken

~ Projektbeschreibung ~

### C. Konzept

#### 1. Lehrsimulation

##### 1.1. Einführung

Ein zentrales Ziel von RUBIN ist es, die Dynamik und Komplexität biologischer Informationsverarbeitung in neuronalen Netzwerken zugänglich zu machen, also Lernenden nicht nur (deklaratives) Faktenwissen, sondern auch prozedurales Wissen zu vermitteln. Hierzu werden Modellsysteme der Neurowissenschaften simuliert und interaktiv visualisiert. Dabei soll ein Fundus an Beispielsimulationen aufgebaut und innerhalb des Simulationslehrsystems (siehe C-2) in einen größeren Kontext gestellt werden. Damit ein hoher Grad an Wiedererkennung gegeben ist, gibt es für die Beispielsimulationen ein einheitliches Format: die Lehrsimulationen.

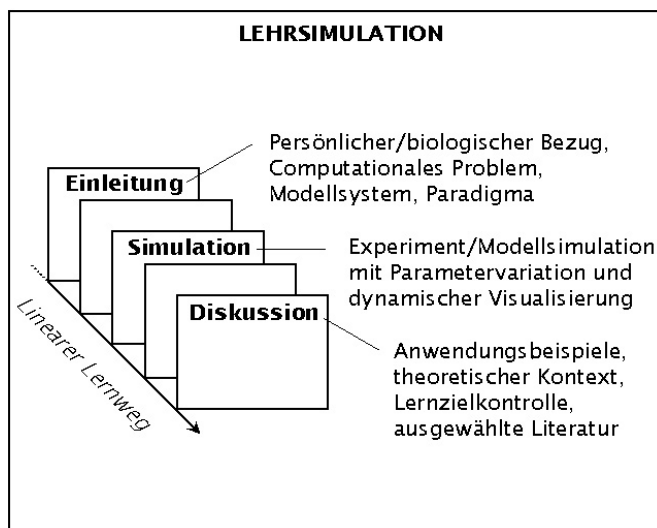


Abb.1 Schematische Darstellung einer Lehrsimulation.

Lehrsimulationen sind didaktisch aufbereitete Beispielsimulationen von Modellen der Neurowissenschaft, die einem generellen Aufbau folgen, alle dieselbe Simulationsumgebung aufweisen und, je nach Inhalt, die verschiedensten interaktiven Visualisierungsmöglichkeiten enthalten. Alle Lehrsimulationen werden zudem durch den Einsatz multimedialer Techniken, wie Video, Audio und Animationen, unterstützt. Der Aufbau der Lehrsimulationen soll helfen, den Bezug der abstrakten Materie formaler Modellsimulation zu real lebenden Systemen herzustellen, um, ausgehend von einem biologischen oder einem psychologischen Beispiel aus der alltäglichen Erfahrungswelt, die Entwicklung und Motivation eines Modells nachvollziehen zu können. Mit der Gliederung *Einleitung – Simulation - Diskussion* orientiert sich eine Lehrsimulation an der üblichen Aufteilung von Lehrtexten oder wissenschaftlichen Veröffentlichungen, ohne jedoch textlastig zu sein. Da Rechner sich nicht besonders gut zum Lesen eignen, wird Lernenden die Möglichkeit gegeben, sich die Texte als auditorische Komponente 'vorlesen' zu lassen.

## 1.2. Interaktive Visualisierung

Interaktive Visualisierung ist das zentrale mediendidaktische Prinzip in RUBIN.

Alles, vom neurowissenschaftlichen Modellsystem (dem Neuron, Netzwerk oder Organismus) bis zur Datendarstellung der Simulation, ist graphisch umgesetzt und jedes graphische Element ist für Aktionen der Lernenden (Mausrolllover, Mausklicks) offen. Jede Aktion der Lernenden wird sofort vom Programm registriert und dynamisch (in Laufzeit) visualisiert. Entscheidend ist hierbei, daß die Visualisierungen von Modellsimulation und Daten parallel dargestellt werden.

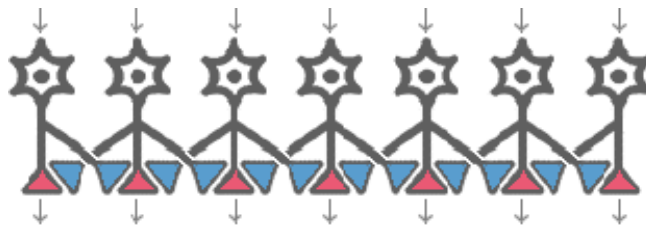
Dies heißt zum Beispiel, daß Aktionen von Lernenden, wie die Verstellung eines Schiebereglers, sich sofort in der graphischen Modellrepräsentation niederschlagen (Teile ändern ihre Farbe) und sich parallel dazu die Datenvisualisierung (z.B. ein einfacher Plot) ändert, so daß verschiedene Beschreibungsebenen oder Stufen der Informationsverarbeitung gleichzeitig bewegt dargeboten werden können. Im Prinzip sind der Kombination von Visualisierungen keine Grenzen gesetzt, so daß sich im Extremfall durch eine einzige Mausaktion etliche Bereiche des Bildschirms, jeder einer anderen Logik folgend, bewegen. Der Schlüssel zum Verständnis der Modelle liegt jedoch in der Wahl der für einen bestimmten Sachzusammenhang geeigneten Kombination der Visualisierungstechniken.

Im folgenden wird geschildert, welche interaktiven Visualisierungen entwickelt wurden und wie sie eingesetzt werden können, um die Dynamik und Komplexität neuronaler Informationsverarbeitung lernbar zu machen.

### 1.2.1. Simulationsumgebung

Den Kern der Lehrsimulationen bildet die Simulationsumgebung. Ihre wesentlichen Bestandteile sind eine graphische Repräsentation des jeweiligen Modells, ein Neuroid zur Konfiguration einzelner Neurone eines Modells und eine Steuerungsfläche (siehe Abb.4) zur Bedienung der Simulation.

Lernende benötigen eine graphische Repräsentation des in einer Lehrsimulation erklärten Modells (siehe Abb.2).



Repräsentation einer Schicht von Neuronen mit lateraler 2-Nachbar Verschaltung. Rote Flächen sind positive „Gewichte“, blaue Flächen negative.

Abb.2

Solche oder so ähnliche Darstellungen können dazu dienen, Modellverschaltungen zu symbolisieren. Entscheidend ist, daß die einzelnen Elemente der Modellrepräsentation von Lernenden konfiguriert werden können. In der graphischen Repräsentation eines Netzwerkes heißt das, daß jedes Neuron per Mausklick anwählbar ist. Auf einen Klick erscheint eine schematische Repräsentation des angewählten Elements, mit der sich dieses dann individuell konfigurieren läßt.

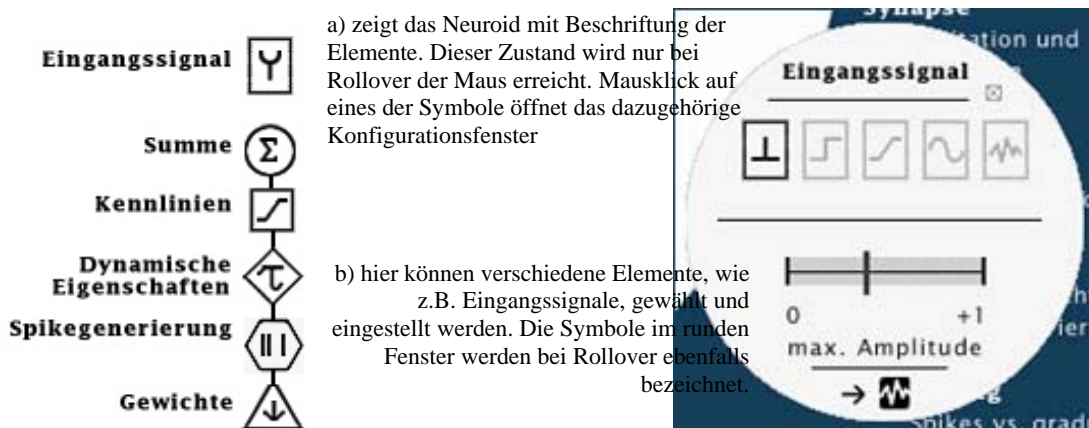


Abb.3

Diese schematische Repräsentation wird *Neuroid* genannt, da es ein modelliertes Neuron darstellt und selbst bei detaillierter Modellierung niemals einem realen Neuron entsprechen kann. Um das auch in Texten klar herausstellen zu können, ist es sinnvoll, dem modellierten Neuron einen eigenen Namen zu geben. Die schematische Darstellung des Neuroids (siehe Abb.3a) ist symbolorientiert. Neurone können auf verschiedenen Abstraktionsniveaus modelliert werden. Die Grundlage des Neuroids bildet das Standardmodellneuron aus der Neuroinformatik, jedoch werden dem einige Elemente hinzugefügt, die der kybernetischen Neurobiologie entlehnt sind. Die einzelnen Symbole repräsentieren die einzelnen Funktionsteile eines Neurons.

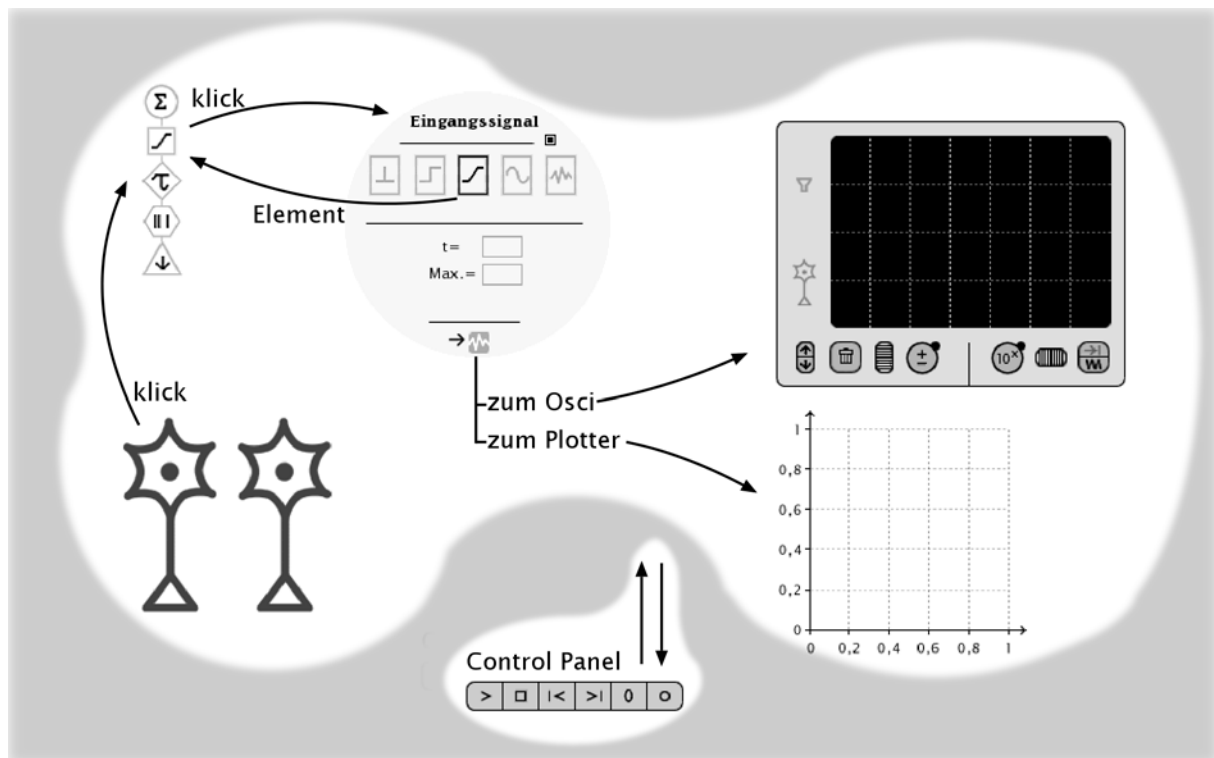


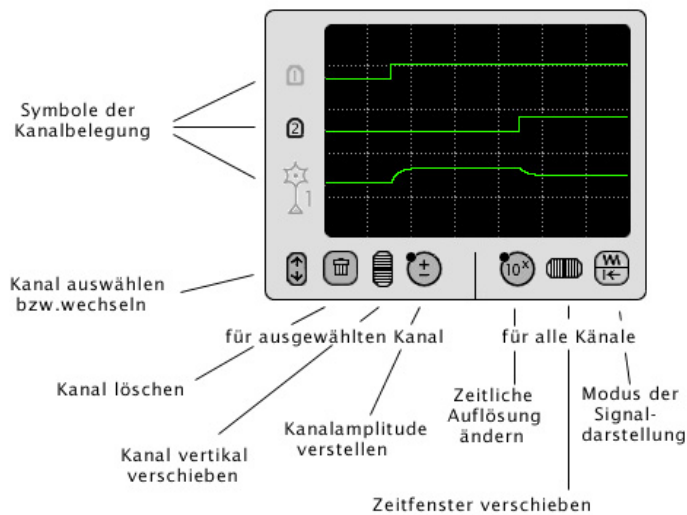
Abb.4 Steuerung der Simulation: Die wichtigsten Interaktionen zwischen den einzelnen Simulationselementen sind hier dargestellt. Ausgehend von dem graphischen Neuron kann über das Neuroid das Parameterfenster angefordert werden, um einzelne Modellteile des Neuroids zu konfigurieren. Von hier aus können auch einzelne Verarbeitungsschritte an die Datenvisualisierungselemente weitergeleitet werden. Mit Oszilloskop und Plotter sind hier zwei Standardelemente zur interaktiven Visualisierung aufgeführt. An dieser Stelle könnten auch andere Visualisierungselemente, wie z.B. ein 3D-Plotter oder eine falschfarbencodierte Pixeldarstellung (siehe Text) angesprochen werden. Einzelheiten zu den Simulationselementen siehe Text.

Mit dem Neuroid können einzelne Neurone der graphischen Repräsentation konfiguriert werden. Klicken die Lernenden auf eines der Symbole, öffnet sich ein Konfigurationsfenster (siehe Abb.3b), in dem Variationen von z.B. Eingangssignalen oder zeitlichen Filtern gewählt und bestimmte Parameter, wie Amplitudenstärke oder der Wert einer Zeitkonstante, variiert werden können. Das Neuroid ermöglicht damit ein hohes Maß an Interaktivität zur Veränderung der vorgegebenen Modellsimulationen, wodurch Lernende in die Lage versetzt werden, individuelle Einstellungen vorzunehmen und die Wirkung ihrer Änderungen unmittelbar durch die Laufzeitvisualisierung beurteilen zu können.

Die Simulation bedarf einer zentralen Steuerung (siehe Abb.4: 'Control Panel'). Hier kann ein Durchlauf gestartet, gestoppt und 'vor- und zurückgespult' werden. Ferner können die Voreinstellungen der jeweiligen Lehrsimulation wieder geladen oder alle Parameter auf Null gesetzt werden. Lernende haben also die Möglichkeit in der Simulationszeit zu navigieren und vorgenommene Änderungen rückgängig zu machen. Welche Aktionen von Lernenden im Speziellen durchzuführen sind, um an die gewünschten Ergebnisse zu kommen, wird zusätzlich durch textbasierte Instruktionen erreicht. Diese Instruktionen sind in die vorhandenen Textblöcke (siehe 1.3) der Lehrsimulationen eingebettet.

### 1.2.2. Oszilloskop

Das Standardgerät zur Visualisierung zeitabhängiger Signale ist das Oszilloskop. Es kann bis zu fünf Signale gleichzeitig darstellen und ist mit den wichtigsten Bedienmöglichkeiten ausgestattet. Einzelne Kanäle können über die symbolische Signalrepräsentation (Neuroid) angefordert oder über die Mülltonne auf der Bedienleiste des Oszilloskops gelöscht werden (siehe Abb.5). Jedes Signal ist individuell in Baseline und Amplitude verstellbar. Für alle Kanäle können Zeitfenster und Zeitauflösung variiert werden. Lernenden stehen zudem zwei verschiedene Darstellungsmodi, kontinuierlich und getriggert, zur Verfügung.



Das Oszilloskop mit Beschriftung der einzelnen Bedienelemente. Rollover über eines der Elemente zeigt seine Funktionsbezeichnung. Erläuterungen siehe Text.

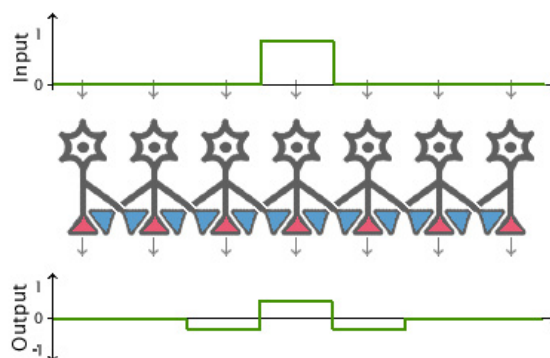
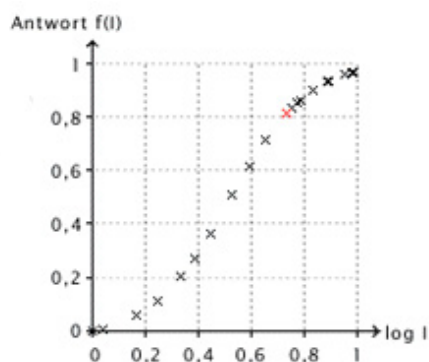
Abb.5

Das Oszilloskop ist ein Gerät aus dem neurobiologischen Alltag. Hier können Unerfahrene am simulierten Oszilloskop die Bedienung der Standardfunktionen lernen und ein Gefühl für die Signalmanipulation bekommen. Genau wie im realen Oszilloskop können Signalamplituden in der Simulation aus dem dargestellten Bereich des Oszilloskops verschwinden und müssen von Lernenden durch die Veränderung der Signaldarstellung wiedergefunden werden.

Eine sinnvolle Zusatzfunktion des Oszilloskops ist die Möglichkeit, eine akustische Signalversion zu erzeugen. Diese Funktion wird in die nächste Version des Oszilloskops integriert werden.

### 1.2.3. Plotter

Eine universellere Darstellungsform ist der Plot (siehe Abb.6). In 2D- und 3D-Version eignet er sich zur Funktionendarstellung und zur (raum-zeitlichen) Datenvisualisierung. Der Plotter kann in verschiedenen Zusammenhängen und in verschiedener Form eingesetzt werden. Der Plotter hat eine eingeschränkte Anwenderbedienung. In den meisten Fällen ist die Einstellung vorgegeben und an das Simulationsbeispiel angepaßt. Bei einer 3D-Variante können Lernende den Plot drehen und zoomen.



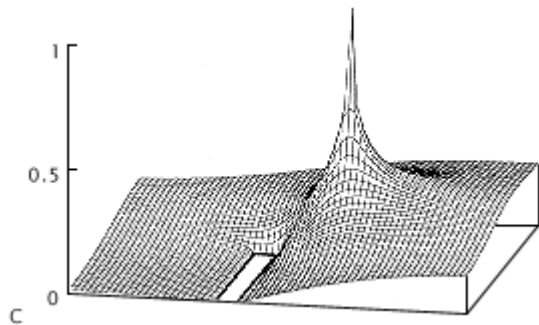


Abb.6

Plottervariationen in RUBIN

a) Standardplotter, hier mit einer sigmoiden Kennlinien;  
 b) Plotter für Eingangs- und Ausgangssignal im Kontext der Netzrepräsentation, hier das Beispiel laterale Inhibition,  
 c) 3D-Plotter, hier mit Plot der Aktivität eines zwei-dimensionalen Neuronengitters zur Berechnung des Diffusionsalgorithmus (Kindermann et al. 1996)

1.2.4. Pixeldarstellung und Falschfarbenkodierung

Einfache Pixeldarstellungen mit Falschfarbenkodierung (siehe Abb.8), oder 2D-Arrays der räumlichen Aktivierung in Netzwerken (siehe Abb.7) können die Anschauung enorm verbessern. Jedes Pixel symbolisiert die Aktivität eines Neuroids. Durch Aneinanderreihung verschiedener Zeitpunkte entsteht so eine Darstellung der raum-zeitlichen Aktivitätsmuster innerhalb eines Netzwerkes. Besonders vorteilhaft ist die parallele Verwendung mehrerer

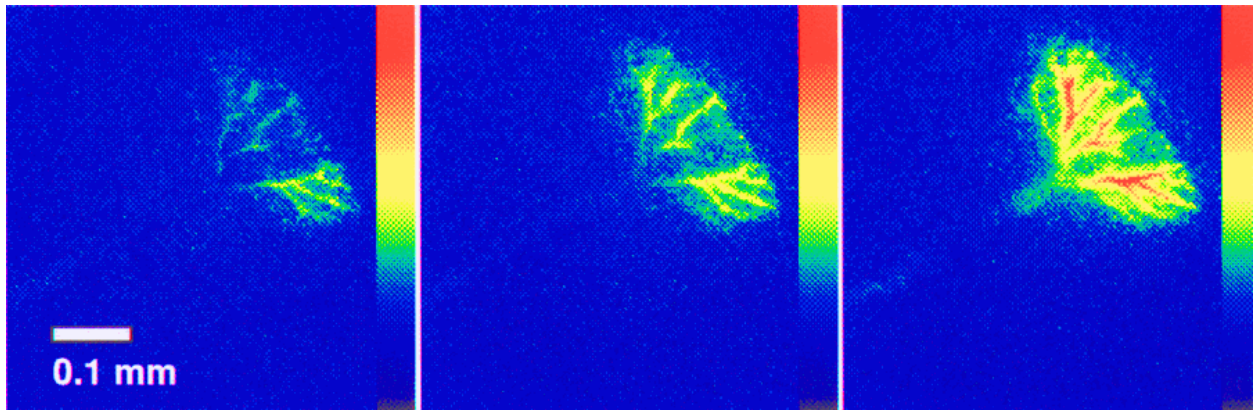


Abb.8 Beispiel für eine Bildreihe mit Falschfarbenkodierung. Gezeigt wird der Calciumeinstrom in den Dendriten einer Nervenzelle über drei Zeitschritte. Der Calciumeinstrom dient als Meßgröße für synaptische Aktivität.

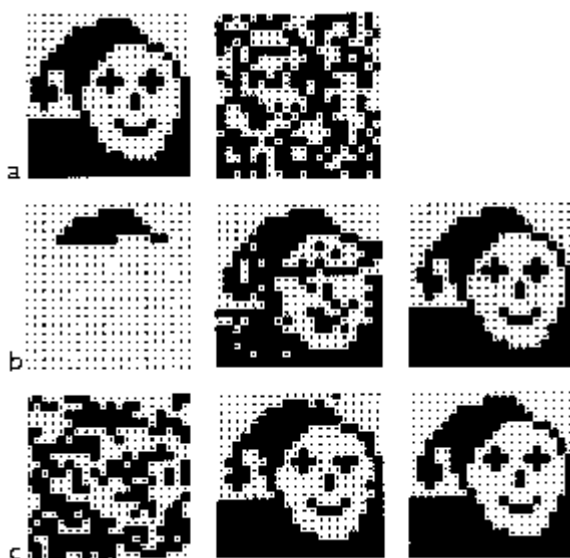


Abb.7

Beispiel für eine Pixeldarstellung aus der Mustererkennung. Gezeigt ist die Aktivität über zwei Zeitschritte eines Hopfieldnetzes, bestehend aus 20x20 Neuroiden, die je zwei Zustände, schwarz und weiß, annehmen können. Die Gewichte des Netzes können 20 verschiedenen Muster speichern.

- a) Ein Muster ist durch das Gesicht repräsentiert, die anderen 19 sind Zufallsmuster.
- b) Zeigt man einen Teil des Gesichtsmusters als Eingabe, kann nach bereits zwei Iterationen das Muster rekonstruiert werden.
- c) Hier wurde das komplette Gesichtsmuster angelegt, jedoch in einer verrauschten Version. Auch hier wird das Muster bereits nach zwei Iterationszyklen rekonstruiert. (Ritter et al. 1992)

Diese Art der Visualisierung von Netzaktivität ist ein typisches Beispiel für eine einfache und anschauliche raum-zeitliche Visualisierung, die direkt an eine Simulation gekoppelt werden kann.

Pixeldarstellungen, um gleichzeitig die Aktivität verschiedener Netzwerkschichten zeigen zu können. In gleicher Form können zum Beispiel Vektorfelder, wie sie im Falle der optischer Flußfelder entstehen, visualisiert werden (siehe Abb.10).

### 1.2.5. Interaktive Animationen

Wenn die Animation von Objekten an die Simulationsdaten gekoppelt ist, können Interaktionen von Lernenden einen direkten Einfluß auf Animationen haben.

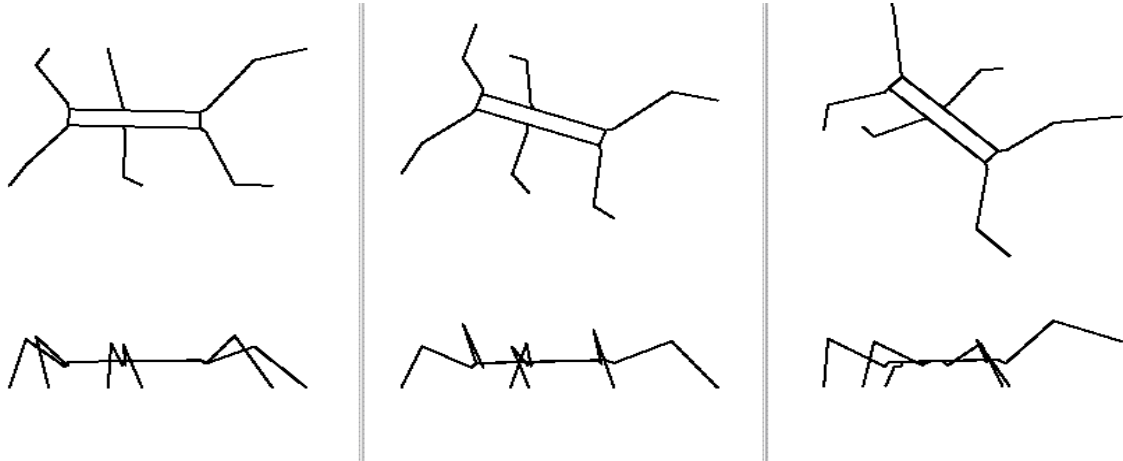


Abb.9 Visualisierung der Laufmaschine Walknet, basierend auf einer Simulation der Gangs einer Stabheuschrecke; Das 'Tier' wird aus zwei Perspektiven gezeigt. Der Gang kann über den Mauszeiger von Lernenden gestört werden, beispielsweise kann das Tier zu Fall gebracht werden.

Wenn z.B. Lernende in der Lehrsimulation 'Laufen der Stabheuschrecke' Übertragungseigenschaften einzelner Neuroide ändern, können sie die Animation 'zu Fall bringen'. Auch eine direkte Interaktion mit der Animation (über den Mauszeiger) ist vorstellbar (siehe Abb.9)

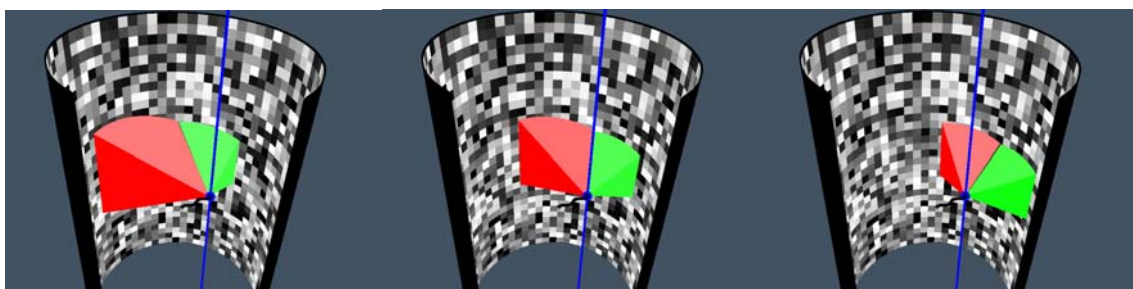
Schon vorhandene biologische Datensätze, die nicht durch eine Simulation erzeugt werden, können ebenfalls in Form animierter Sequenzen gezeigt werden.

Dem Einsatz von Animationen sind prinzipiell keine Grenzen gesetzt. Dies kann bis zu kleinen 'Comicsequenzen' führen oder einem »virtuellen Tier« (siehe auch 3.2).

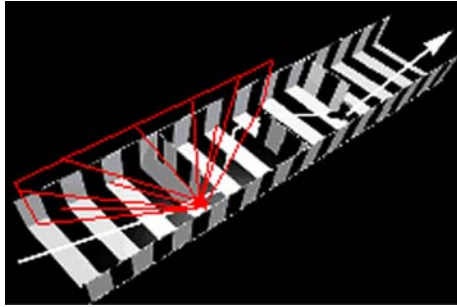
### 1.2.6. Virtuelle Räume und 3D-Animation

Der Einsatz von virtuellen Räumen und 3D-Animationen nimmt innerhalb von RUBIN bislang keinen Raum ein. Allerdings liegen auch hier Verwendungsmöglichkeiten auf der Hand. Beinbewegungen von Insekten können als 3D-Animation gezeigt und interaktiv bewegt oder gestört werden. Es ist ferner vorstellbar, z.B. ein Modell visueller Bewegungsdetektion durch einen virtuellen Raum mit beliebigen Texturen fliegen zu lassen, um so die Wirkung (quasi-)natürlicher Reize im Unterschied zu Standardreizen testen zu können (siehe Abb.10). Aber auch zur Exploration anatomischer oder physiologischer Zusammenhänge können virtuelle Räume sehr nützlich sein.

Abb.10



a) Zylindrische Arena mit Zufallstextur. Der rote Bereich zeigt das Sichtfeld des linken Auges eines Modells des visuellen Systems der Fliege. Der grüne Bereich dementsprechend das Sichtfeld des rechten Auges. Der blaue Punkt beschreibt die Position des Modellsystems. Die blaue Linie beschreibt die Hochachse, um die sich Modell im Raum drehen läßt und somit seine Blickrichtung ändert. (Lutterklas 1999)



b) Hindernisarena mit Modellsystem (roter Punkt) und Sichtfeld des linken Auges (rote Linien). Pfeil zeigt Flugbahn.

a) bis f): Beispiele für eine 3D-Visualisierung im Virtuellen Raum.  
 a) (oben) und b) (links) zeigen verschiedene virtuelle Arenen aus der Beobachterperspektive. Die farbigen Bereiche markieren die Sichtfelder des Modellsystems.  
 c) zeigt die Perspektive des Modellsystems in Arena b) (linkes Auge, rot). (Lutterklas 1996)  
 Abbildungen d)-f) (unten) visualisieren die Aktivität der verschiedenen Verarbeitungsstufen des Modellsystems von d) der Retina über e) eine Neuronenschicht, die Eingangssignale zeitlich Bandpaß filtert zu f) einer Schicht von lokalen Bewegungsdetektoren, die als Gesamtausgabe (grüne Pfeile) ein optisches Flußfeld liefern. (Lorenz 1996)



c) Eingabemuster; d) Aktivierung der Retina; e) zeitliche Bandpaßfilterung; f) optisches Flußfeld

Lernende können durch die Virtuellen Räume navigieren, Objekte aus verschiedenen Perspektiven (z.B. Beobachter- oder Tierperspektive) betrachten und gegebenenfalls verschiedene Tiefenebenen anschauen. Simulationen können auch direkt auf eine virtuelle Umwelt rückgekoppelt werden. Dieser gekoppelte Einsatz erfordert allerdings einen hohen Rechenaufwand und ist daher im Einzelnen abzuwägen.

### 1.2.7. Formeln

Verschiedenen Funktionsteile werden Farben zugeordnet, so daß Variablen (wie Eingang, Gewicht, Ausgang) schon aufgrund ihrer Farbgebung identifiziert werden können. Formeln können, ähnlich wie Grafiken, direkt an die (Inter-)Aktionen der Lernenden gekoppelt werden. So können die tatsächlich eingestellten Werte in einer Formel erscheinen, sobald die Parameter eingestellt wurden. Mit der Simulation ändern die Variablen in der Formel ihre Werte entsprechend des Simulationsverlaufs.

Es soll erreicht werden, die Lernenden zu den Formeln zu bringen, nicht die Formeln zu den Lernenden. Daher ist das Motto: 'Konzepte vor Formeln'.

### 1.3. Fließtext

Die kurzen, aber immer verfügbaren Texte dienen als Bindeglieder zwischen den Bildschirmseiten einer Lehrsimulation. Die Einführungs- und Diskussionsteile enthalten Informationen über den Kontext der Modellsimulation, im Simulationsteil kommen Hinweise für die Interaktionsmöglichkeiten hinzu. Diese Informationen formieren einen linearen Lernweg und machen die Simulation zu einer Lehrsimulation. Der lineare Lernweg ist graphisch durch die am Textende befindlichen Navigationspfeile gekennzeichnet. Wenn die Lernenden also dem linearen Lernweg folgen wollen, ist die Aufmerksamkeit der Lernenden automatisch auf den Fließtext gerichtet, da die Positionierung der Navigationspfeile auf der Bildschirmseite vom Text abhängen. Wenn sie einen anderen Lernmodus benutzen (z.B. explorativ), stehen den Lernenden andere Navigationsmöglichkeiten zur Verfügung (siehe C-2.5.1). Außerdem bieten Hypertext-Links Möglichkeiten, Exkurse zu machen (Glossareinträge, Links zu anderen Lehrsimulationen, Internet-Adressen) oder Originaltexte zu finden (Literaturhinweise).

Im Allgemeinen wird die Verwendung von Text jedoch auf ein Minimum reduziert, da sich der Bildschirm nicht gut zum Lesen von Fließtext eignet. Aus diesem Grund wird auch vollständig auf Rollbalken verzichtet.

Längere Texte werden in kleine Textblöcke aufgeteilt und den Lernenden auf mehreren aufeinanderfolgenden Bildschirmseiten gemeinsam mit Abbildungen dargeboten.

#### 1.4. Unterstützende Medien

Für die Bereiche einer Lehrsimulation, die keine interaktive Visualisierung benötigen, sondern bei denen eine vorgefertigte Darstellung von Sachverhalten und Zusammenhängen ausreicht, kann ein Film oder eine Animation eingesetzt werden. Lernphasen ohne Interaktivität (Pausen) können der Effektivität und Motivation zuträglich sein. So können zum Beispiel Verhaltensleistungen von Tieren als Animation oder als Film gezeigt werden. Graphiken und Fotos illustrieren den Text und Blockschaltbilder dienen der Veranschaulichung größerer Zusammenhänge. Es ist ferner vorgesehen, den Fließtext als Audiodatei bereitzustellen, so daß Lernenden ihn sich 'vorlesen' lassen können und der Leseaufwand am Bildschirm minimiert wird.

#### 1.5. Lernzielkontrolle

Eine automatische Lernzielkontrolle vereint in sich mehrere Vorteile des rechnergestützten Unterrichts. Die automatische Bewertung der beantworteten Fragen unterstützt die Lernenden in der Selbstkontrolle des neu erworbenen Wissens und fördert den Prozeß des Wissenserwerbs im Selbststudium.

Durch Zuordnungsaufgaben (Ja/Nein Alternative, Multiple-Choice, Anordnungen und Zuordnungen) und alphanumerische Eingaben (Text und/oder Zahlen, Lückentext) soll das zum Verständnis der Modellsimulationen nötige deklarative Wissen abgefragt werden.

Das Verständnis von Modellen, wie die hier interessierenden Modelle neuronaler Informationsverarbeitung, sind wegen ihres prozeduralen Charakters der automatischen Bewertung schwerer zugänglich. Eine Form der Bewertung des Modellverständnisses ergibt sich bereits aus der Variation der Modellparameter und der Einschätzung von deren Richtigkeit über die Visualisierungen (Learning by doing). Weitere Formen der Lernzielkontrolle sind in den Ausbaustufen vorgesehen (siehe C-3.3)

## 2. Simulationslehrsystem

### 2.1. Motivation

Die einzelnen Lehrsimulationen sind aufeinander aufbauend in ein Simulationslehrsystem eingebettet (siehe auch Abb.17). Wie bereits ausführlich beschrieben (siehe C-1) besitzen die einzelnen Lehrsimulationen eine in sich geschlossene Struktur, die sie einzelstehend verwendbar und verständlich macht. Anspruchsvollere Lehrsimulationen jedoch sind für LERNENDE OHNE BASISWISSEN nicht nachzuvollziehen. Ein großer Teil der Zielgruppe würde verfehlt (z.B. Biologen im Grundstudium, Kognitionswissenschaftler, Schüler, Lehrer). Für diesen Teil der Zielgruppe müßte in anspruchsvolleren Lehrsimulationen entweder der inhaltliche Kontext eines Modells oder die Simulationstechniken so umfassend eingeführt werden, daß die vorgesehene Länge (15-30 Minuten Bearbeitungszeit) für die Lehrsimulationen bei weitem überschritten würde. Anspruchsvolle Lehrsimulationen können also nur gelehrt oder gelernt werden, wenn sie in einen umfassenden Lernweg eingebettet sind, durch den das Basiswissen vermittelt wird.

Das inhaltliche Basiswissen (vornehmlich deklarativ) kann im Verlauf einer Vorlesung, eines Seminars bzw. über herkömmliche Lehrbücher erworben werden, erfordert also kein Simulationslehrsystem. Kritisch für die Notwendigkeit eines linearen Lernwegs ist das Basiswissen der Simulationstechniken (vornehmlich prozedural): Dieses Basiswissen kann weder in Vorlesungen oder Seminaren noch in herkömmlichen Lehrbüchern erworben werden. Die Lernenden können sich das Basiswissen der Simulationstechniken nur aneignen, wenn sie ein didaktisches Werkzeug für den Computer zur Verfügung haben, das sie begleitend zu den von ihnen besuchten Veranstaltungen einsetzen können. Für das Selbststudium benötigen sie über die Einführung in das Basiswissen Simulationstechnik hinaus noch eine Einführung in das inhaltliche Basiswissen. Wie ein Simulationslehrsystem aussehen sollte, das diese Eigenschaften in sich vereint, sei im folgenden beschrieben.



## 2.2. Gliederung

Das Simulationslehrsystem besteht aus einzelnen Lehrsimulationen. Die Lehrsimulationen sind zwar einzelstehend, z.B. für Präsentationen, zu verwenden, sind jedoch zusätzlich so konzipiert, daß sie aufeinander aufbauen und, im Zusammenhang gesehen, einen linearen Lernweg bilden (siehe auch Abb.17). Die inhaltliche Kohärenz zwischen den Lehrsimulationen wird durch eine zusätzliche Gliederungsebene unterstützt: Zwei oder mehr Lehrsimulationen formieren einen Themenkomplex, der eine eigene inhaltliche Einführung besitzen kann. Diese ist ein multimedial unterstützter Lehrtext (ohne Simulation), in dem ein Bezug zu vorauslaufenden und nachfolgenden Lehrsimulationen und Themenkomplexen aufgezeigt wird (siehe Abb.11).

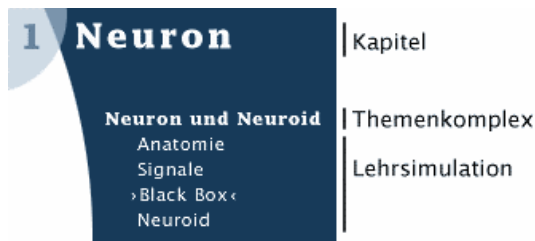


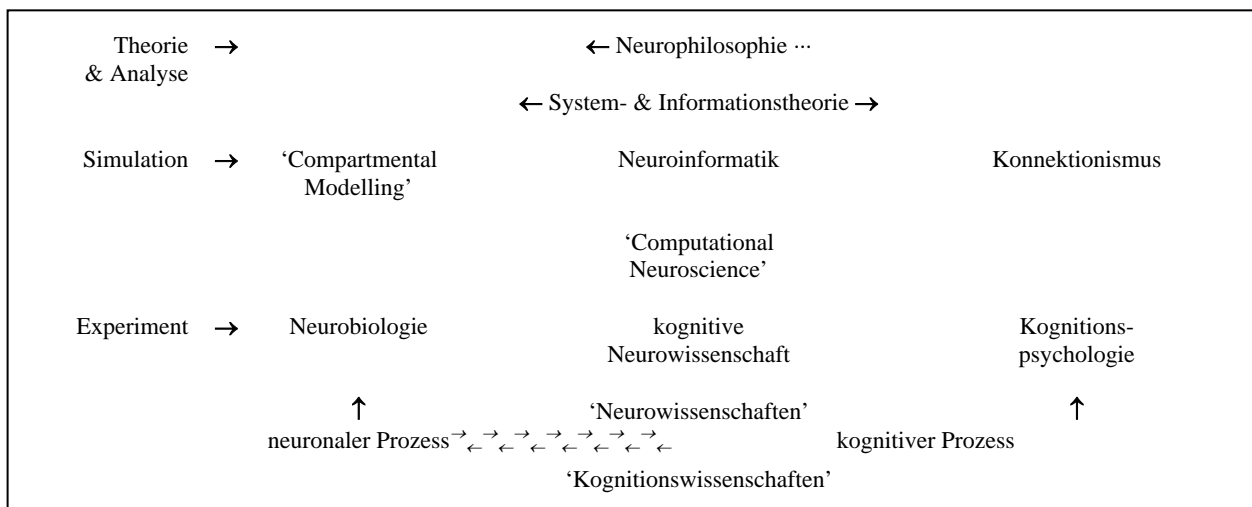
Abb.11

Gliederungsebenen in RUBIN.

Mehrere Themenkomplexe bilden ein Kapitel. Die fünf vorgesehenen Kapitel beschreiben einen linearen Lernweg, der »bottom up« von einfachen zu komplexen Inhalten und Simulationstechniken verläuft. Im folgenden wird das Wissen, das Lernende im Simulationslehrsystem erwerben können, getrennt nach Wissen über Inhalte und Wissen über Simulationstechniken erläutert.

## 2.3. Inhalte

Von inhaltlicher Seite betrachtet, enthalten die Lehrsimulationen konzeptuelles (deklaratives) Wissen der Neurowissenschaften und angrenzender Disziplinen (siehe Abb.12 ). Bei der hierfür vorgesehenen Gliederung (siehe Abb.13) des Simulationslehrsystems fällt auf, daß sie sich an neurobiologischen Konzepten orientiert. Dies hat zwei Gründe: Erstens stellt die Neurobiologie das naturwissenschaftlich-experimentelle Fundament für eine reduktionistische Erklärung der informationsverarbeitenden Leistungen des Gehirns dar, und zweitens bietet die Neurobiologie den Lernenden im Gegensatz zu den anderen beteiligten formaleren Disziplinen einen anschaulichen Bezug zu ihrer alltäglich erlebten Welt: Jeder weiß, was ein Tier, ein Auge, eine Zelle ist, aber



nicht jeder Lernende weiß, was eine Kennlinie, eine Fourier-Transformation, oder Transinformation ist. Die einzelnen Kapitel stellen verschiedene Organisations Ebenen (und Simulationstiefen) neurobiologischer Prozesse dar. Ausgegangen wird vom Neuron als zentrale Informationsverarbeitende Einheit des Nervensystems. Molekularbiologie, Biochemie und Biophysik (inkl. »Compartmental Modelling«) werden nicht behandelt, da

sie nicht viel zu der rein funktionalistischen Beschreibung von Neuronen beitragen, die für das Verständnis der Lehrsimulationen entscheidend ist. Die Netzwerke stellen die nächsthöhere Organisationsebene neuronaler Informationsverarbeitung dar.

Abb.12 Schema zur Einordnung der einzelnen Disziplinen.

## Aufbau des Simulationslehrsystems

Hier werden anhand von neurobiologischen Modellsystemen verschiedene Verschaltungsprinzipien und die resultierenden neuen Informationsverarbeitungsebenen aufgezeigt. Die Kapitel Sensorik und Motorik beziehen sich auf eine erneut höhere Organisationsebene (Funktionssysteme), in der die Grundprobleme 'Filterung' und 'Steuerung' bearbeitet werden. Im letzten Kapitel wird schließlich die organismisch-individuelle Beschreibungsebene geklärt, die das Verhalten in die Lage versetzt, adäquaten Informationsverarbeitung unter den Fragestellungen der sensomotorischen Kopplung und der Generierung adäquaten Verhaltens integriert zu betrachten (C-3.2.). Konzeptuelles Wissen der anderen Disziplinen (siehe Abb.12) wird in den Einführungs- und Diskussionsteilen der Lehrsimulationen eingebunden oder erarbeitet und sind über das Glossar (siehe C-2.5.2.) abrufbar.

Das Neuron ist die zentrale informationsverarbeitende Einheit im Nervensystem. Dieses einführende Kapitel umfasst die wichtigsten Verrechnungen, die einzelne Neurone im Nervensystem leisten.

Sinnessysteme befähigen den Organismus Informationen aus seiner Umwelt aufzunehmen und zu verarbeiten. Die Prinzipien und Probleme sensorischer Informationsverarbeitung werden mit exemplarischen Modellsimulationen erklärt.

Wie tragen Neurone, Netzwerke, Sensorik und Motorik dazu bei, dass Tiere sich in ihrer Umwelt angepasst verhalten können? Am Beispiel von Modellen zum Verhalten von Tieren und Menschen werden die Inhalte in allgemeinerer Form rekapituliert.

1. Neuron	2. Netzwerk	3. Sensorik	4. Motorik	5. Verhalten
<p>Neuron und Neuroid</p> <ul style="list-style-type: none"> <li>▪ Anatomie des Neurons</li> <li>▪ Signale</li> <li>▪ Black-Box</li> <li>▪ Neuroid</li> </ul> <p>Verrechnung</p> <ul style="list-style-type: none"> <li>▪ Kennlinien</li> <li>▪ Reaktion und Reiz</li> </ul> <p>Synapse</p> <ul style="list-style-type: none"> <li>▪ Exzitation und Inhibition</li> </ul> <p>Zeitverhalten</p> <ul style="list-style-type: none"> <li>▪ Zeitliche Filter</li> <li>▪ Phasisch und tonisch</li> <li>▪ Adaptation</li> </ul> <p>Spikes</p> <ul style="list-style-type: none"> <li>▪ Alles oder Nichts</li> <li>▪ Spike-Generierung</li> </ul> <p>Kodierung</p> <ul style="list-style-type: none"> <li>▪ Spikes vs. Graduiert</li> <li>▪ Rate vs. Timing</li> <li>▪ Zuverlässigkeit</li> </ul>	<p>Konvergenz und Divergenz</p> <ul style="list-style-type: none"> <li>▪ Räumliche Summation</li> <li>▪ Rezeptive Felder</li> </ul> <p>Laterale Interaktionen</p> <ul style="list-style-type: none"> <li>▪ Laterale Inhibition</li> <li>▪ Bewegungs-detektion</li> </ul> <p>Rückkopplung</p> <ul style="list-style-type: none"> <li>▪ Regelkreise</li> <li>▪ Muster-erzeugung</li> <li>▪ Reafferenz-prinzip</li> </ul> <p>Spezielle Funktionen</p> <ul style="list-style-type: none"> <li>▪ Winner-Take-All</li> <li>▪ Optimierung</li> <li>▪ Assoziatives Speichern</li> </ul> <p>Kodierung #2</p> <p>verteilt vs. integriert</p>	<p>Reizaufnahme</p> <ul style="list-style-type: none"> <li>▪ Reize</li> <li>▪ Transduktion</li> <li>▪ Transformation</li> </ul> <p>Sinnessysteme</p> <ul style="list-style-type: none"> <li>▪ Mechanisch</li> <li>▪ Optisch</li> <li>▪ Chemisch</li> <li>▪ Elektrisch</li> <li>▪ Magnetisch</li> <li>▪ Multimodal</li> </ul> <p>Modellsysteme</p> <ul style="list-style-type: none"> <li>▪ Bewegungs-sehen derFliege</li> <li>▪ Olfaktorik der Biene</li> <li>▪ Richtungshören der Eule</li> <li>▪ Echoorientierung der Fledermaus</li> <li>▪ Elektrischer Fisch</li> </ul>	<p>Motorische Einheit</p> <ul style="list-style-type: none"> <li>▪ Motoneurone</li> <li>▪ Innervation</li> <li>▪ Elektromech. Kopplung</li> <li>▪ Einzelzuckung</li> </ul> <p>Kontrollsysteme</p> <ul style="list-style-type: none"> <li>▪ Lokale Kontrolle</li> <li>▪ Reflexe</li> <li>▪ CPGs</li> <li>▪ Programme</li> </ul> <p>Modellsysteme</p> <ul style="list-style-type: none"> <li>▪ Fluchtreaktion des Blutegels</li> <li>▪ Laufen der Stabheuschrecke</li> </ul>	<p>Adäquates Verhalten</p> <ul style="list-style-type: none"> <li>▪ Senso-motorische Integration</li> <li>▪ Verhalten und Umwelt</li> <li>▪ Verhalten und Evolution</li> </ul> <p>Modellierung</p> <ul style="list-style-type: none"> <li>▪ Neuronale Systeme</li> <li>▪ Meßdaten und Modelle</li> <li>▪ Heuristik</li> <li>▪ Künstliche Systeme</li> </ul> <p>Biol. Informationsverarbeitung</p> <ul style="list-style-type: none"> <li>▪ Information</li> <li>▪ Methoden</li> <li>▪ Analyseebenen</li> </ul>

Abb.13 Inhaltlicher Aufbau des Simulationslehrsystems. Die Gliederung der einzelnen Kapitel ist vorläufig. Weitere Erläuterungen siehe Text.



Abb.14 Die RUBIN-Startseite und die unterschiedlichen Oberflächen für die fünf Kapitel.

In Hinblick auf das Verständnis der Modellsimulationen ist entscheidend, daß konzeptuelles Wissen, aber auch Formelwissen aus der Neuroinformatik (Theorie neuronaler Netzwerke), der Informationstheorie und der Systemtheorie/Kybernetik vermittelt wird, da diese Disziplinen die zum Verständnis notwendigen formalen Beschreibungs-, Analyse- und Bewertungsmethoden bereitstellen.

Wissen aus den Kognitionswissenschaften fließt in drei Formen in das Simulationslehresystem ein: Erstens in Form von konzeptuellem Wissen der Kognitionspsychologie (Modelle von Wahrnehmung, Gedächtnis, Aufmerksamkeit etc.) und hybriden Ansätzen, wie der kognitiven Neurowissenschaft, das an entsprechenden Punkten des Lernwegs integriert wird (z.B. Gedächtnis bei Lernen), zweitens in Form von psychophysischen Bildschirmexperimenten (Demonstration psychophysischer Phänome, optischer Täuschungen, Gedächtnistests) und schließlich, drittens in Form von konnektionistischen Modellsimulationen kognitiver Prozesse (Gedächtnisleistung), die formal gesehen letztendlich die gleichen Annahmen machen, die auch den neuroinformatischen Simulationen zugrundeliegen.

Globales Ziel des Simulationslehrsystems in Bezug auf das zu vermittelnde konzeptuelle oder inhaltliche Wissen ist, Lernenden ein solides Basiswissen der Neurowissenschaften zu vermitteln und die Lernenden darüber hinaus in die Lage zu versetzen, Gemeinsamkeiten und Unterschiede der verschiedenen Ansätze in den Neurowissenschaften zu überblicken und Konzepte ihrer eigenen Disziplin (sei es Biologie, Informatik oder eine kognitionswissenschaftliche Disziplin) mit Konzepten anderer Disziplinen abgleichen zu können. Ziel in Bezug auf die Simulationstechnik ist, die in den Simulationen behandelten Modelle in einen inhaltlichen biologischen, informatischen oder kognitionswissenschaftlichen Kontext und in die Alltagswelt einbetten zu können.

## 2.4. Simulationstechnik

Eine entscheidende Besonderheit von RUBIN liegt darin, daß außer dem zuvor beschriebenen konzeptuellen oder deklarativen Wissen ein zumindest gleichgewichtiges Maß an prozeduralem Wissen vermittelt wird, das in keiner anderen Form als einem Simulationslehrsystem gelehrt werden kann. Das prozedurale Wissen bezieht sich auf den Umgang mit neurowissenschaftlichen Modellsimulationen und wird unter dem Begriff Simulationstechnik subsumiert und im folgenden Abschnitt beschrieben.

Betrachtet man das erste Kapitel von der Seite der zu lernenden Simulationstechniken, ist sein Ziel vor allem, mit dem »Neuroid«, der schematischen Modellversion des Neurons, und seinen vielfältigen interaktiven Möglichkeiten sowie der daran gekoppelten dynamischen Visualisierung umzugehen (siehe C-1). Die Kohärenz zwischen den Lehrsimulationen wird dabei über eine einheitliche Symbolik und Graphik hergestellt: Das Neuroid ist immer vollständig sichtbar. Im ersten Kapitel werden die Interaktionsmöglichkeiten immer nur ausschnittsweise zugelassen. Diesem Prinzip der eingeschränkten Anwenderaktion folgend, sind die irrelevanten Teile jedoch nicht aktivierbar. Die Lehrsimulationen sind auf die in der jeweiligen Lehrsimulation zu erklärende Informationsverarbeitungsleistung vorkonfiguriert und nur festgelegte Ausschnitte des Paramterraums sind frei konfigurierbar, so daß die Lernenden Schritt für Schritt an den freien Umgang mit dem Neuroid herangeführt werden. Dieses instruktionalistische Vorgehen ist notwendig, weil die Erfahrung mit professionellen Simulationsumgebungen zeigt, daß sich Lernende von den vielen Interaktionsmöglichkeiten 'erschlagen' fühlen, und ein effektives Lernen nicht zustande kommt. Gleichzeitig lernen Lernende durch den direkten Bezug zur realen Welt, der in Einführung und Diskussion hergestellt wird, nicht einfach abstrakt 'ins Blaue' zu simulieren, sondern die Simulation auf konkrete Ziele auszurichten. (Eine Simulation um der Simulation willen macht für die Zielgruppe keinen Sinn!) Das Lernziel für das erste Kapitel im Bereich Simulationstechnik ist also der sichere Umgang mit dem Neuroid und seinen Interaktions- und Visualisierungsmöglichkeiten. Konkret heißt das, die Lernenden sollten die Symbole des Neuroids und der gekoppelten Fenster und Objekte erkennen und benennen können und wissen, welche Mausaktionen zu welchen Ergebnissen führen.

Im zweiten Kapitel wird die Kenntnis des Neuroids vorausgesetzt. Mit den aus Neuroiden bestehenden Netzwerken wird die nächste Komplexitätsstufe erreicht. Das Ziel ist nunmehr, zu verstehen, welche neuen Verrechnungen möglich werden, wenn Informationen parallel und verteilt verarbeitet werden. Da das einzelne Neuron nicht mehr im Vordergrund steht, wird das Neuroid, wenn möglich, nur in einer reduzierten Form dargestellt. Wie im ersten Kapitel bildet es jedoch weiterhin die Basis für die Interaktionen. Bei einer graphischen Repräsentation eines Netzwerkes beispielsweise, lassen sich Neurone mit der Maus aktivieren und das entsprechende Neuroid wird, wie im ersten Kapitel, interaktiv zugänglich, so daß Netzwerkeigenschaften verändert werden können. Je nach Beispiel können Lernende so ganze Schichten oder Netzwerke über ein Neuroid steuern oder sogar jedes einzelne Neuron im Netzwerk für sich konfigurieren (siehe Abb.4). Nach Bearbeitung des zweiten Kapitels sollten Lernende in der Lage sein, kleinere Netzwerke ohne voreingestellte Parameter so zu konfigurieren, daß sie eine spezifische Informationsverarbeitungsleistung vollziehen können. Die ersten beiden Kapitel enthalten das Basiswissen der Simulationstechnik, das die Lernenden befähigt, auch anspruchsvollere Simulationen zu verstehen. Das Ziel der folgenden drei Kapitel Sensorik, Motorik und Verhalten ist, dieses Wissen zu konsolidieren und flexibel auf verschiedenste Modellsysteme anzuwenden. Grundsätzlich neu im Hinblick auf die Simulationstechnik ist hier, daß neue Komplexitätsstufen der Systeme erreicht und damit weitere Informationsverarbeitungsleistungen und Analysemethoden behandelt werden.

## 2.5. Weitere Eigenschaften des Simulationslehrsystems

### 2.5.1. Benutzerführung

Von der Startseite aus können Lernende direkt in das gewünschte Kapitel gelangen. Wenn sich die Lernenden auf den kapitelspezifischen Seiten (siehe Abb.14) befinden, stehen ihnen bereits umfangreiche Navigationmöglichkeiten zur Verfügung (Abb.15). Alle anderen Kapitel sind ebenso über einen Mausklick zu erreichen, wie jede in diesem Kapitel befindliche Lehrsimulation und das Hauptmenü. Innerhalb der Lehrsimulationen ist eine textboxorientierte Navigation vorgesehen (siehe C-1.3), um sich auf dem linearen Lernweg innerhalb der Lehrsimulation oder zwischen Lehrsimulationen vorwärts und rückwärts zu bewegen. Darüber hinaus existiert eine globale Navigation (entspricht dem 'Vor'und 'Zurück' in Internetbrowsern), über die die Lernenden die zuvor erschienenen Seiten und damit ihren persönlichen Lernweg nachvollziehen können, auch wenn sie den linearen Lernweg verlassen haben. In der Vorwärtsbewegung können sie anstelle der Textboxnavigation auch die globale Navigation nutzen, um im linearen Lernweg voranzuschreiten. Über das Hauptmenü können weitere Module, beispielsweise das Glossar erreicht werden.

## 2.5.2. Multimediales Glossar

Neben der graphischen Aufbereitung der Gliederung, der den Lernenden auch einen ansatzweise explorativen Zugang zum Wissen ermöglicht, wird über das Glossar ein direkter lexikalischer Zugang bereitgestellt (siehe Abb.16). Ein multimedialer Glossareintrag besteht aus einer Bildschirmseite mit einer textuellen Beschreibung, den Verbindungen zu den entsprechenden Lehrsimulationen und Schaltflächen sowie einer Sammlung aller zu diesem Thema vorhandener Medien (Graphik, Animationen etc.). Die Glossarseiten sind über Hyperlinks auch direkt aus den Texten der einzelnen Lehrsimulationen zu erreichen. Zurück zu der zuvor bearbeiteten Lehrsimulation gelangt die Lernenden über die globale Navigation.

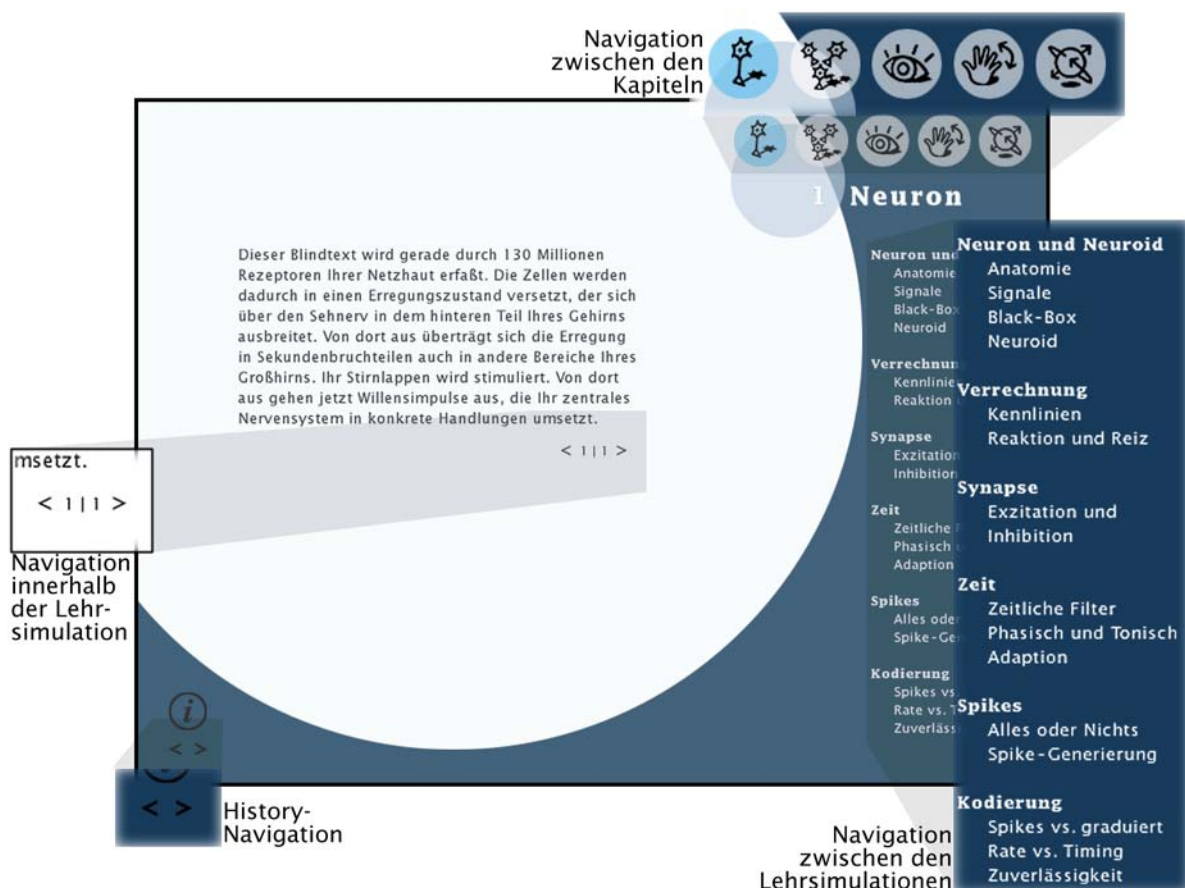


Abb.15 Oberfläche des ersten Kapitels. Die Elemente zur Navigation sind hervorgehoben.

## 2.5.3. Hilfesystem

Wenn sich die Bilschirmelemente nicht selbst erklären, wie es z.B. bei den speziellen Bedienelementen der Simulationsumgebung der Fall ist, besitzen sie eine Rolloverfunktion, durch die automatisch die textuelle Bezeichnung des Elements erscheint. In anderen Fällen, in denen eine längere textuelle Beschreibung für Funktionen von Bilschirmelementen notwendig ist, können die Lernenden zusätzlich über das Hauptmenü eine Hilfefunktion aktivieren, die einige erläuternde Sätze enthält (Sprechblasen). Die gesamte Funktionalität und der Aufbau des Simulationslehrsystems sind Bestandteil des Anwenderhandbuchs. Ein 'Online'-Anwenderhandbuch (Einführungskapitel) ist bereits in einer prototypischen Version implementiert. Optional besteht die Möglichkeit, eine Printversion des Anwenderhandbuchs zu erstellen.

~ Zwischenbilanz ~

Die Lehrsimulationen und das Simulationslehrsystem stellen in der oben beschriebenen Form ein komplettes Paket dar. Vom didaktischen und funktionalen Aspekt entspricht es allerdings in dieser Form nicht dem Standard, der z.T. heute bereits etabliert ist und sich im Laufe der nächsten Jahre etablieren wird. Daher werden im folgenden noch weitere Module beschrieben, die RUBIN zu einem Lern- und Lehrsystem mit Modellcharakter machen würden.

### 3. Lernumgebung

#### 3.1. Motivation

Die hier aufgeführten Module erweitern die Funktionalität von RUBIN vornehmlich in Hinblick auf die Lerneffektivität, den Lernkomfort sowie die Benutzerfreundlichkeit und stellen zum Teil entscheidende Vorarbeiten für spätere Ausbaustufen dar (siehe C-4,C-5). Die Entwicklung der Lernumgebung ist zwar konzeptionell und zeitlich vom Produktionsprozess zu entkoppeln, stellt jedoch einen integralen Bestandteil des RUBIN-Konzepts dar, wie bereits durch die im Hauptmenü vorgesehenen Schaltflächen angedeutet ist (siehe Abb.16).

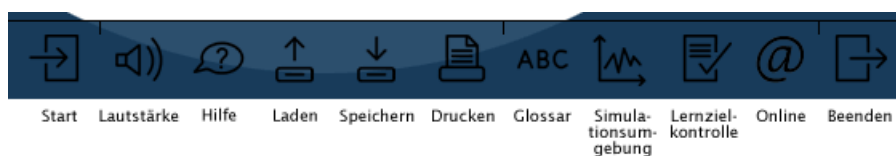


Abb.16 Das zuladbare Hauptmenü, das der Erreichung zusätzlicher Funktionen und der Navigation zwischen Modulen dient.

#### 3.2. »Virtuelles Tier«

Das virtuelle Tier ist ein mit den im Simulationslehrsystem vermittelten Simulationstechniken modelliertes Programmmodul, das eine animierte graphische Gestalt hat und fähig ist, bestimmte Verhaltensweisen auszuführen. Beispielsweise bewegt es sich über den Bildschirm, wendet sich Lichtquellen zu, sucht 'Futterquellen' oder findet 'seinen Weg nach Hause'.

Zu Beginn des Lernwegs können die Lernenden mit dieser graphischen Repräsentation des virtuellen Tieres interagieren und seine Fähigkeiten kennenlernen. Hierdurch wird den Lernenden unmittelbar beim Einstieg in den Lernweg deutlich gemacht, worum es in der gesamten Anwendung geht: zu erklären, wie neuronale Informationsverarbeitung Tiere und Menschen befähigt, sich sinnvoll in einer gegebenen Umwelt zu verhalten. Diese Funktion des virtuellen Tieres wird als didaktische Klammer bezeichnet.

Zur Ausführung seines Verhaltens benötigt das virtuelle Tier Neuronen in Netzwerken in den Bereichen Sensorik und Motorik. In oder nach jedem Kapitel können die Lernenden mit dem entsprechenden Simulationsteil des virtuellen Tieres interagieren und unmittelbar beobachten, welche Auswirkungen ihre Änderungen in der Simulation auf das Verhalten des virtuellen Tieres haben. Hierbei kann es sein, daß das Tier

sich nicht mehr sinnvoll verhält oder plötzlich andere Verhaltensweisen entwickelt. Ob Lernende ihr Lernziel erreicht haben oder nicht, können sie zwar punktuell auch in der automatischen Lernzielkontrolle der einzelnen Lehrsimulationen erfahren (siehe C-1.5). Ob sie allerdings fähig sind, das Wissen, das sie beispielsweise in einem ganzen Kapitel gelernt haben, zu überblicken und anzuwenden, können sie nur erfahren, wenn sie komplexere und anwendungsbezogene Aufgaben bearbeiten. Die Aufgaben, die den Lernenden über das virtuelle Tier gestellt werden, stellen also eine globale Lernzielkontrolle dar, die Lernende selbst überprüfen können.

Um eine Art Handlung oder Spannungsbogen aufzubauen, wird den Lernenden zu Beginn des Lernwegs zusätzlich die konkrete Aufgabe gestellt, dem virtuellen Tier im Verlauf des Lernwegs (im Zusammenhang mit der globalen Lernzielkontrolle), etwas beizubringen, was es zu Beginn noch nicht kann; beispielsweise zu lernen, den Mauszeiger des Lernenden zu fixieren und ihm zu folgen.

Das virtuelle Tier ist also eine Metapher, die drei Funktionen gleichzeitig erfüllt. Es dient als didaktische Klammer, globale Lernzielkontrolle und Träger eines Spannungsbogens.

### 3.3. Benutzerprofil

Lernende unterscheiden sich voneinander. Um dieser Tatsache in Ansätzen gerecht zu werden, ist ein Modul vorgesehen, das Daten des beschrittenen Lernwegs speichert, visualisiert und zum Teil auch bewertet. Zu Beginn des Lernwegs können sich Lernende hierfür unter einem Namen registrieren. In jeder Sitzung können die bereits gewonnenen Daten wieder geladen werden, wenn sich Lernende mit ihrem Namen registrieren. Folgende Daten sollen durch das Programmmodul erhoben werden:

Besuchen einer Seite, Verweildauer auf der Seite, Interaktionen (z.B. Abruf von Medien, Anzahl der Klicks auf interaktiven Objekten wie Neuroid oder Oszilloskop), Reihenfolge der Seiten bzw. Lehrsimulationen sowie der Lernerfolg und die Lernerfolgsentwicklung in der automatischen Lernzielkontrolle.

Lernende können sich so über ihr Lernverhalten informieren und die Lernwegvisualisierung und Lernwegauswahl (siehe C-3.4/5) nutzen.

Teile der erhobenen Daten können zu Zwecken einer summativen Evaluation (siehe C-3.6) oder zu einer Beurteilung der Lernenden durch Lehrende verwendet werden (siehe C-5.4).

Um bei jeder neuen Sitzung auf die Daten der vorausgegangenen Sitzungen zurückgreifen zu können, wird die Möglichkeit bereitgestellt, an einer beliebigen Stelle des Lernwegs bzw. am Ende der Sitzung automatisch zu speichern bzw. Daten erneut hinzuzuladen.

Lernende können eine Sitzung genau an der Stelle im Lernweg beginnen, an der sie die letzte Sitzung beendet haben. Lehrende können sich so beispielsweise eine spezielle Parameterkonfiguration in einer Lehrsimulation 'zurechtlegen', die sie in einem Vortrag verwenden wollen und müssen nicht erneut alle Einstellungen während des Vortrags vornehmen. Die Möglichkeit, Daten (des Lernwegs etc.) zu speichern und zu einem anderen Zeitpunkt wieder zu laden, stellt einen entscheidenden Vorteil von 'echten' Anwendungen gegenüber allen Browserbasierten Anwendungen dar, da der Zugriff auf die Festplatte aus dem Browser heraus nach wie aus Sicherheitsgründen nicht möglich ist.

### 3.4. Lernwegvisualisierungen

Im Simulationslehrsystem ist vornehmlich der lineare Lernweg konzipiert und realisiert worden (siehe aber C-2.5.1), da sich z.B. explorative Lernwege ohne weiteres vom linearen Lernweg ableiten lassen, aber nicht umgekehrt. Bei der Bereitstellung eines vollkommen explorativen Lernwegs muß man das Problem berücksichtigen, daß Lernende den Überblick über ihre Lernwege verlieren ('Lost in Hyperspace'). Um einen vollkommen explorativen Lernweg bzw. explorative Seitenäste didaktisch sinnvoll bereitstellen zu können, muß er strukturiert dargestellt werden. Die Lernenden benötigen eine übersichtliche Lernwegvisualisierung. Die Inhalte der gesamten Anwendung sollen hierzu in symbolischer Form auf einer Bildschirmseite dargestellt werden. Prinzipiell soll jede Bildschirmseite über maximal zwei Mausklicks zu erreichen sein. Bereits besuchte Inhalte ('Öffnen der Seite') und bearbeitete Inhalte ('Leistung in der automatischen Lernzielkontrolle') sind für die Lernenden z.B. über eine Farbkodierung erkennbar. In einer zweiten Darstellungsform wird visualisiert, in welcher Reihenfolge Bildschirmseiten besucht wurden und welchen Inhaltsbezug sie hatten. Die Entwicklung der Lernwegvisualisierungen ist eng verknüpft mit der Entwicklung des Benutzerprofils (siehe C-3.3).

### 3.5. Lernwegauswahl

Die Lernenden haben verschiedene Lerngewohnheiten und Vorkenntnisse. Mit der Registrierung eines neuen Lernenden läßt sich elegant die Abfrage der individuellen Lerngewohnheiten und Vorkenntnisse verbinden. In dem Modul Lernwegauswahl soll eine solche Abfrage realisiert werden, auf deren Basis eine Lernweggestaltung vorgeschlagen wird. In der Abfrage werden Daten über Status (Schüler/Student/Graduiert) und Schwerpunkt (Laie, Interessierter, Biologe, Informatiker, Psychologe etc.) erhoben. Um die Vorkenntnisse der Lernenden einschätzen zu können, wird eine Art 'Antestat' entwickelt (z.B. Kennen Sie den Begriff 'Hippocampus', den Begriff 'Kennlinie', den Begriff 'Transinformation'? etc.).

Am Ende der Abfrage wird von dem Programmmodul vorgeschlagen, welcher Lernweg eingeschlagen werden sollte. Es sei kurz erläutert, wie sich verschiedene Lernwege realisieren lassen: Für fortgeschrittene Lernende oder ExpertInnen eignet sich beispielsweise ein explorativer Lernweg, in dem vorgeschlagen wird, die Lernwegvisualisierungen und den lexikalischen Zugang zur Navigation zu nutzen. Einsteigende Lernende sollten einem linearen, instruktionalistisch orientierten Lernweg folgen, wie er in den Lehrsimulationen vorgesehen ist. Hierbei könnte die Regel gelten, daß nur dann eine Lehrsimulation abrufbar ist, wenn der lineare Lernweg zuvor vollständig bearbeitet wurde und alle Lernzielkontrollen positiv waren. Verschiedene fachliche Schwerpunkte lassen sich aus der Reihenfolge und Benennung der einzelnen Lehrsimulationen generieren. Lernende mit formalen Vorkenntnissen und/oder Programmiererfahrung erhalten dann beispielsweise eine Gliederung, die sich an der Systemtheorie orientiert. Lernende mit kognitionswissenschaftlichem Schwerpunkt erhalten beispielsweise anstelle des Kapitels Sensorik ein Kapitel Wahrnehmung mit ausgewählten Lehrsimulationen.

### 3.6. Lern- und Lehrplaner: Evaluation

Um die in RUBIN verwendete Didaktik sowie den Lernerfolg und das Lernverhalten der Lernenden, aber auch das Lehrverhalten der Lehrenden beurteilen zu können, wird eine umfangreichere (vorerst) abschliessende (summative) Evaluation durchgeführt. Auf Basis der in Gruppeneinsätzen sowie über das Benutzerprofil (siehe C-3.3) erhobenen Daten wird ein Lehrplan erstellt, der Empfehlungen über die Charakterisierung der Lernenden(-gruppe) und die Reihenfolge, die Dauer und die Erfolgszielkontrolle des Einsatzes von Lehrsimulationen sowie über begleitende Lehrtätigkeiten enthält. Diese Empfehlungen werden in eine graphische Benutzeroberfläche umgesetzt, in der Lehrende das Lernen vorplanen (Stunden- bzw. Lehrplan) und Gruppen organisieren können, d.h. die Lernenden im Kurs sind aufgeführt und ihre Leistung kann protokolliert werden. Diese Werkzeuge für Lehrende werden besonders im Zusammenhang mit der Online-Nutzung (siehe C-5) und Ausrichtung eines Web-Based-Training (WBT) essentiell. Lernende bekommen die Planungsfunktion zur Verfügung gestellt, um ihr Selbststudium strukturieren zu können.

## 4. Entwicklungsumgebung

### 4.1. Motivation

Die im folgenden beschriebenen Module Simulationswerkzeug und Skripter machen RUBIN zu einem generativen Lern- und Lehrsystem, da sie ihm Eigenschaften eines Autorenwerkzeugs verleihen, mit dem von Lernenden und Lehrenden eigene Lehrsimulationen erstellt werden können. Lernenden wird so die wissenschaftspropädeutisch wertvolle Möglichkeit gegeben, Modellsimulationen in einer intuitiv bedienbaren und durch das Simulationslehrsystem gewohnten graphischen Benutzeroberfläche selbst zu erstellen. Lehrenden, die im universitären Kontext gleichzeitig Wissenschaftler sind, wird es erstens möglich, Lehrsimulationen oder experimentelle Daten nach ihren didaktischen Präferenzen neu zu gestalten oder zu organisieren und zweitens, ihre aktuellen Forschungsinhalte Lernenden oder anderen Wissenschaftlern in einem multimedial und didaktisch aufgearbeiteten Format zu präsentieren.

Das freie Simulationswerkzeug und der Skripter eröffnen, gemeinsam verwendet, Möglichkeiten für Lernende und Lehrende, nicht nur Rezipient, sondern auch Autor zu sein. Dieses Merkmal besitzt nach unserem



Kenntnisstand heute noch kein anderes Lern- und Lehrsystem. Durch die im nachfolgenden Abschnitt geschilderten Erweiterungen der Online-Umgebung (siehe C-5), kann die Funktionalität der Entwicklungsumgebung noch potenziert werden.

#### 4.2. Simulationswerkzeug

Lernende werden durch die in RUBIN zur Verfügung gestellten didaktischen Werkzeuge befähigt, Modellsimulationen zu verstehen und zu steuern. Der letzte Schritt, den Lernende machen müssen, um das erworbene Wissen im wissenschaftlichen Kontext praktisch anwenden zu können, ist, Modellsimulationen selbst zu erstellen. Zu diesem Zweck wird eine graphische Benutzeroberfläche entwickelt, mit der Lernende Modellsimulationen über 'drag and drop' erstellen können. Hierbei sollen die aus den Lehrsimulationen (siehe C-1) bekannten didaktisch aufbereiteten Simulationselemente (Neuroid, Neuronenschemata etc.) und Visualisierungswerkzeuge (Plotter, Oszilloskop etc.) als Baukasten bereitgestellt werden (siehe Abb.4) Die vorerst vorgesehene Funktionalität reicht nicht an die wissenschaftlichen Simulationsprogramme heran, reicht aber aus, um zahlreiche Modellsysteme zu simulieren und zu visualisieren. Ein solches graphisch orientiertes Simulationswerkzeug hat den Vorteil, das es im Gegensatz zu anderen wissenschaftlichen Simulationsprogrammen auch ohne Programmierkenntnisse (gilt für viele Biologen, Kognitionswissenschaftler) angewendet werden kann. Hiermit eröffnen sich vielfältige neue Einsatzmöglichkeiten: Lernende können eine bisher nicht beschriebene Form der globalen Lernzielkontrolle nutzen, indem sie beispielsweise am Ende eines Kapitels die Aufgabe bekommen, eine bestimmte Informationsverarbeitungsleistung zu simulieren, von der sie nur die Eingangs- und Ausgangssignale kennen. Ferner können sie das Simulationswerkzeug nach Belieben aufrufen, eigene Ideen umsetzen und Fragen, die sie sich stellen, selbst beantworten. Auch für Lehrende stellt das freie Simulationswerkzeug eine Möglichkeit dar, individuelle Interessen zu verfolgen. Kein Lehrmedium kann die speziellen Bedürfnisse aller Lehrenden erfüllen. Jeder Lehrende hat seine eigenen Vorstellungen, welche Inhalte vermittelt werden sollen und welche Didaktik dabei zum Einsatz kommt. Solche Vorstellungen können Lehrende mit einem freien Simulationswerkzeug umsetzen, indem sie beispielweise Modelle aus ihrem wissenschaftlichen Alltag mit dem didaktisch aufbereiteten 'Baukastensystem' simulieren und visualisieren, um sie in Vorträgen zu präsentieren.

#### 4.3. Skriptter

Damit Lernende und Lehrende die mit dem freien Simulationswerkzeug erstellten Simulationen dokumentieren können, wird eine Editierfunktion bereitgestellt, mit der Simulationen in einen Lernweg aus Texten und anderen Medien eingebettet werden können und zu einer LEHRsimulation werden. Die Medienunterstützung erhöht den Wirkungsgrad der Simulationen in Vorträgen und ermöglicht darüber hinaus, den Einsatz neu entwickelter Simulationen im Selbststudium. Die Editierfunktion können die Lernenden auch verwenden, um während des Lernens Notizen zu machen und zu speichern.

Zu diesem Zweck wird, basierend auf einer zuvor entwickelten Skriptsprache, eine 'Schablone' entwickelt, in die Texte und Medien sowie die Simulation eingefügt und letztendlich in einem definierten Format gespeichert werden. Eine graphische Benutzeroberfläche mit Auswahl Fenstern, über die ein Zugriff auf lokale Daten (z.B. Gif- oder Jpg-Bilder sowie Texte) ermöglicht wird, soll gewährleisten, daß keine Programmierkenntnisse benötigt werden, um diesen Skriptter zu benutzen.

### 5. Online-Umgebung

#### 5.1. Motivation

Was letztlich für eine weite, aber geregelte Verbreitung der RUBIN-Module fehlt und die durch die Entwicklungsumgebung realisierten generativen Eigenschaften zur Entfaltung bringt, sind Kommunikationsmöglichkeiten.

Zunächst werden Online-Nutzungsmöglichkeiten beschrieben, die nicht unbedingt zentrale Strukturen (Server) erfordern. Das Problem bei serverbasierten Nutzungen ist, das sie ein größeres Maß an Administration, Pflege

und Wartung erfordern als Anwendungen, die von einer lokalen Installation aus gesteuert werden. Für einige Funktionen ist der Aufbau eines Servers jedoch unerlässlich. Diese werden abschließend geschildert.

## 5.2. RUBIN-Kommunikationsschnittstelle

Die modulare und plattformübergreifende Java-Implementierung von RUBIN öffnet ein weites Anwendungsfeld für die Online-Nutzung. Der entscheidende Schritt zur Nutzung dieser Möglichkeiten ist die Implementierung einer von JAVA unterstützten Kommunikationsschnittstelle. Bisher ist von der Situation ausgegangen worden, daß Lernende oder Lehrende mit einer lokal installierten Version von RUBIN arbeiten. Fast alle Computer in Universitäten und auch die meisten Heimrechner verfügen jedoch über einen dauerhaften oder zeitweisen Zugang zum Internet und damit über die Möglichkeit, die Online-Schnittstelle zu nutzen.

## 5.3. Kooperatives Lernen

Kommunikation unter Studierenden fördert Lernmotivation und Lernerfolg. Hierfür ist eine gesonderte Funktion vorgesehen, mit der Lernende z.B. während einer Veranstaltung aus der Lernumgebung heraus miteinander kommunizieren können und die an die Online-Kommunikationsform 'Chat' angelehnt ist. Sie können sich in Gruppenräumen oder über das Internet gegenseitig beraten, wie sie eine Aufgabe bearbeiten können oder sich gegenseitig informieren, daß und wo sie eine Parameterkonfiguration oder selbst erstellte Lehrsimulation gespeichert haben, die sich andere laden können.

## 5.4. WBT-Erweiterung für den Lern- und Lehrplaner

Für die geregelte Durchführung eines Web-Based-Trainings (WBT) benötigen Lehrende zusätzlich eine erweiterte Funktionalität des Lern- und Lehrplaners, mit der verteilte Gruppen von Lernenden online registriert und Informationen über den Ablauf des WBT (Ankündigungen, Aufgaben, Abgabetermine, 'Sprechstunde') veröffentlicht werden können. Bei der Verwaltung ist entscheidend, daß die Benutzer individuell berücksichtigt werden. Die elektronische Lehre und die damit verbundene Automatisierung der Erfolgskontrolle eröffnet Lernenden vielfältige Möglichkeiten, eine Bearbeitung der ihnen gestellten Aufgaben zu umgehen, z.B. indem sie sich Musterlösungen besorgen. Wenn die Lernenden individuell betreut werden und jeder eine andere Aufgabe (z.B. zufallsgenerierte Parameterkonfigurationen) bekommt, könnten solche Probleme teilweise umgangen werden.

Eine Kommunikation zwischen Lehrenden und Lernenden ist ferner erforderlich, um eine nichtautomatische, 'persönliche' Lernzielkontrolle durch Lehrende zu ermöglichen, da sich Modellsimulationen z.T. aufgrund ihres großen möglichen Zustands- und Lösungsraums einer automatischen Bewertung entziehen.

Parametereinstellungen von Lehrsimulationen, die mit dem freien Simulationswerkzeug und Skripten erstellt wurden, müssen von Lehrenden überprüft und korrigiert werden können. Hier kommen die bereits beschriebenen Laden- und Speichern-Funktionen zum Tragen, da sie ermöglichen, die (Lehr-) Simulationen oder Parametereinstellungen via Email zu verschicken.

## 5.5. Server

Um die vielfältigen Möglichkeiten einer Online-Nutzung bündeln und organisieren zu können, ist es erforderlich, einen Server aufzubauen. Der Server stellt selbstverständlich auch die WWW-Seiten bereit, über die Lernende und Lehrende auf Serviceleistungen, Updates, News, Links etc. zugreifen können. Die wichtigsten weiteren Nutzungsmöglichkeiten seien kurz beschrieben.

### 5.5.1. Administration

Eine Struktur zur Verwaltung der Anwenderdaten regelt den Zugang zum Server, ermöglicht eine organisierte Online-Distribution der RUBIN-Module und eine Registrierung der AnwenderInnen oder Gruppen. In der

RUBIN-GUI werden entsprechende Abfragefenster bereitgestellt, die eine elektronische Kommunikation und einen einfachen Datenaustausch zwischen AnwenderInnen und AdministratorInnen ermöglichen.

#### 5.5.2 Datenbank

Um die mit dem freien Simulationswerkzeug und Skripten erstellten Lehrsimulationen einer öffentlichen Bewertung und Nutzung zugänglich zu machen, sollten sie auf einem Server archiviert werden können. Da nur dokumentierte Simulationen wirklich aussagekräftig sind, ist vorgesehen, Datenbankstrukturen bereitzustellen, die es ermöglichen bzw. erfordern, Dokumente im Format der Lehrsimulationen abzulegen. Hierdurch wird ein Standard vorgeschlagen, den Lehrende verwenden können, wenn sie Aufgaben durch Lernende bearbeiten lassen. Eine einheitliche Form vereinfacht ferner das Verständnis aller zuvor archivierten dokumentierten Simulationen und erhöht damit ihre Wiederverwertbarkeit. Gleichzeitig steht ein Archiv der in RUBIN verwendeten Beispielsimulationen (und evtl. der verwendeten Medien) zur Verfügung.

Analog zu der Simulationsdatenbank wird eine Datenbank mit dokumentierten biologischen Meßdaten aus Physiologie- und Verhaltensexperimenten eingerichtet. Diese biologischen Datensätze sollen in ähnlicher Weise wie die Modellsimulationen dokumentiert bereitgestellt werden. Dies eröffnet die Möglichkeit, Simulationen durchzuführen, die auf experimentellen Datensätzen beruhen. Die Meßdaten aus Experimenten mit biologischen Systemen dienen der Modellierung als Orientierung am biologischen Vorbild und somit der Modellevaluation. Die Kombination von Daten und Modellen in einer Simulation erlaubt eine forschungsnahe Lehre und untermauert die biologische Plausibilität der verwendeten Modelle.

Diese Strukturen dienen als Archiv und erweitern sich über die Jahre von selbst.

# RUBIN

Rechnergestützter Unterricht zur  
Biologischen Informationsverarbeitung in neuronalen Netzwerken

~ Projektbeschreibung ~

## D. Realisierung<sup>1</sup>

### 1. Basisversion: Simulationslehrsystem

Der im folgenden beschriebene erste Produktionsabschnitt basiert auf den im noch laufenden Förderungszeitraum realisierten Vorarbeiten. Unter Berücksichtigung dieser Vorarbeiten und der momentan vorhandenen oder einer vergleichbar eingearbeiteten personellen und technischen Infrastruktur ist der Produktionsabschnitt Simulationslehrsystem in 4 Mannjahren innerhalb eines Kalenderjahres abzuwickeln.

#### 1.1. Vorarbeiten

Bereits realisiert vom Simulationslehrsystem sind: das konzeptionelle (inhaltliche und didaktisch-funktionale) Grundgerüst (1MJ), das graphische Grundgerüst (0.5MJ) und das technische Grundgerüst (1MJ). Darüberhinaus ist eine technische und personelle Infrastruktur aufgebaut worden, die darauf ausgelegt ist, Drehbücher für Inhalte (Lehrsimulationen) schnell in diesem Grundgerüst zu realisieren.

Die Vorarbeiten seien kurz detaillierter beschrieben:

In den Bereich der inhaltlichen und didaktisch-funktionalen Konzeption fällt die Entwicklung der Lehrsimulationen (siehe C-1) anhand lern- und kognitionspsychologischer Kriterien (inkl. Design der Interaktionen und der funktionalen Oberfläche), alle Autorentätigkeiten (Prototypenentwicklung, Drehbuchvorlage) sowie die Erstellung eines Konzepts für einen linearen Lernweg (Sichtung, Modularisierung und Gliederung des interdisziplinären Wissensgebietes sowie Design der funktionalen Oberfläche). Maßgabe bei dieser Aufgabe war, ein modulares Format zu entwickeln, daß von verschiedenen Autoren benutzt werden kann und programmertechnisch spezifizierbar ist. In den Bereich der Graphik fällt die Umsetzung der funktionalen Oberflächen in graphische Benutzeroberflächen für Lehrsimulationen und das Simulationslehrsystem. Maßgabe bei dieser Aufgabe war insbesondere, einen 'Baukasten' an Graphikobjekten zu entwickeln, der von AutorInnen und ProgrammiererInnen (selbstverständlich auch GraphikerInnen) über die gesamte Projektlaufzeit verwendet werden kann. In den Bereich der Programmierung fällt die Implementierung der Benutzeroberflächen und interaktiven Möglichkeiten von Lehrsimulation und Simulationslehrsystem, die Entwicklung eines Simulationsprogramms, die Entwicklung dynamischer Visualisierungswerkzeuge (Plotter, Oszilloskop, animierte Graphik, Schalter, Regler etc.) sowie die Implementierung von Automatisierungsroutinen von Autorentätigkeiten und elektronischen Veröffentlichungsvorgängen. Maßgabe bei der Aufgabe war die Bereitstellung einer softwaretechnisch einheitlichen, objektorientierten (modularen) und plattformunabhängigen Entwicklungs- und Programmumgebung in JAVA.

---

<sup>1</sup> Bei der Berechnung des Produktionsaufwands wurden eingearbeitete Personen und eine vorhandene technische Infrastruktur vorausgesetzt. Bei einer eventuellen Abwicklung über nur kurzzeitig anwesende studentische Hilfskräfte, DiplomandInnen, StaatsexemansanwärterInnen etc. kann sich der Entwicklungszeitraum z.T. erheblich vergrößern. Die Sachmittel wurden vorerst nicht berücksichtigt, stellen aber erfahrungsgemäß nur einen Bruchteil der Finanzierung dar.

## 1.2. Produktionsaufwand

Das Simulationslehrsystem ist inhaltlich lediglich exemplarisch umgesetzt. Der verbleibende Gesamtaufwand der Produktion des Simulationslehrsystems, wie es bereits ausführlich beschrieben wurde (siehe C-2), umfaßt die 50 Lehrsimulationen sowie 20 Einführungsmodul ohne Simulationsseiten mit einem durchschnittlichen Umfang von jeweils 10 Seiten (siehe Abb.13). Hierbei besitzt jede Seite einen Text, der auch aus einer Audiodatei 'vorgelesen' werden kann, und mindestens ein anderes Medium, sei es Graphik, Film, Animation, o.ä. Ferner sind das multimediale Glossar (siehe C-2.5.2) und die Lernzielkontrollen der einzelnen Lehrsimulationen (siehe C-1) in den Produktionsprozeß des Simulationslehrsystems integriert.

Unter Berücksichtigung der Vorarbeiten und der momentan vorhandenen oder einer vergleichbar eingearbeiteten personellen und technischen Infrastruktur ist der Projektteil Simulationslehrsystem in 4 Mannjahren innerhalb eines Kalenderjahres abzuwickeln. Die einzelnen Bereiche seien im folgenden kurz erläutert.

### 1.2.1. Autorentätigkeiten (ges. 12MM)

Die Erstellung einer Printversion des Drehbuchs für alle Lehrsimulationen des Simulationslehrsystems beinhaltet das Verfassen der Texte (mit Hyperlinks zu anderen Lehrsimulationen, Glossareinträgen und Literaturhinweisen) und die Spezifikation der Graphiken, der möglichen Benutzerinteraktionen mit den Simulationen und Visualisierungswerkzeugen, der Lernzielkontrollen, der Navigationsmöglichkeiten sowie die Teil-Drehbücher für Filme und Animationen (12MM).

### 1.2.2. Medienproduktion (ges. 12MM)

Da die graphischen Elemente der Benutzeroberflächen bereits vorhanden sind, müssen nur noch einzelne im Drehbuch spezifizierte Graphiken (Schemata, Fotos etc.) erstellt werden (3MM). Der Hauptaufwand liegt in der Produktion der Audiodaten (3MM), aber vor allem in der Produktion der Filme, Animationen und virtueller Räume (6MM). Da viele Visualisierungen (Graphik, Animation, virtuelle Räume) direkt mit den Graphikfähigkeiten von JAVA realisiert werden können, greifen die Bereiche Graphik und Programmierung direkt ineinander.

### 1.2.3. Programmierung (ges. 12MM)

Aufgabe der Programmierung ist, die im Drehbuch spezifizierte Funktionalität unter Einbindung der produzierten Graphik in das vorhandene technisch-graphische Grundgerüst in JAVA zu implementieren (9MM). Besonders aufwendig zu produzieren sind die Simulationsseiten, die sich durch die stark interaktive Graphik auszeichnen. Ihr Implementierungsaufwand übersteigt eine 'normale' Bildschirmseite mit Text und Graphik um das vielfache, wird aber durch das Vorhandensein der programmtechnischen Infrastruktur erheblich reduziert. Ferner gehören zu den Aufgaben die Umsetzung der im Drehbuch spezifizierten Module Lernzielkontrolle (1MM) und multimediales Glossar (2MM).

### 1.2.4. Qualitätssicherung (ges. 6MM)

In diesen Bereich fällt die technische Qualitätssicherung, aber auch die formative Evaluation der Didaktik. Im einzelnen sind dies: Die Organisation, Durchführung und Evaluation von experimentellen Einzelsitzungen mit Lernenden zur Qualitätssicherung der Didaktik (3MM) sowie die Organisation, Durchführung und Evaluation von Kurzeinsätzen in Lehrveranstaltungen zur Qualitätssicherung der technischen Funktionalität und Einsetzbarkeit (3MM).

### 1.2.5. Koordination (ges. 6MM)

Unter administrative Tätigkeiten fallen hier alle Aufgaben der Projektorganisation und Außendastellung. Hierzu gehören: Die Bereitstellung einer HTML-basierten Internet-Struktur für eine Online-Nutzung und

Modifikation des Drehbuchs durch die Autoren- und Produktionsteams (3MM), die Koordination der Autoren bei der Produktion des Drehbuchs (1MM), die Akquisition bereits vorhandener Medien und (JAVA)-Sources aus der wissenschaftlichen Forschung (1MM) sowie die Außendarstellung des Projekts (Präsentationen auf Fachtagungen, Erstellung einer umfassenden WWW-Präsenz mit Demoversion etc.)(2MM)

## 2. Ausbaustufen: Lern-, Entwicklungs- und Online-Umgebung

Der im folgenden beschriebene zweite Produktionsabschnitt zerfällt in die drei auch unabhängig voneinander betrachtbaren Ausbaustufen Lern-, Entwicklungs- und Online-Umgebung. Aufgrund der stärker ausgeprägten Modularisierung der einzelnen Projektteile wird der Produktionsaufwand – im Unterschied zu der Beschreibung im ersten Produktionsabschnitt – nicht in Tätigkeitsfelder unterteilt, sondern pro Modul angegeben.

### 2.1. Vorarbeiten

Für die hier beschriebenen Projektteile existieren zum jetzigen Zeitpunkt keine konkreten Vorarbeiten. Da jedoch bereits in der Konzeption des Projekts und der Prototypentwicklung die spätere Integration der Module berücksichtigt wurde, bestehen alle Voraussetzungen für eine programmtechnisch saubere Integration. Darüber hinaus stehen zu Beginn des zweiten Produktionsabschnitts, also nach der vorauslaufend beschriebenen Produktion des Simulationslehrsystemes sehr viele Programmteile zur Verfügung, auf die bei der Produktion der weiteren Module zurückgegriffen werden kann. Nur so kann der z.T. sehr geringe Produktionsaufwand erreicht werden.

### 2.2. Produktionsaufwand

#### 2.2.1. Lernumgebung (ges. 17MM)

Die Entwicklung des Moduls Virtuelles Tier besteht aus

- der Programmierung des 'Animats' (2MM),
- dem Design und der Programmierung der 5 Übungsmodule (1MM),
- der Erstellung der graphischen Benutzeroberflächen und animierten Repräsentation des virtuellen Tieres (1MM).

Die Entwicklung des Moduls Benutzerprofil umfaßt

- das Design und die Programmierung von Abfragen des Lernenden sowie Routinen für die Abfragen des Programmablaufs und deren Speicherung. (2MM)

Die Entwicklung des Moduls Lernwegvisualisierung besteht aus dem

- Design der graphischen Benutzeroberflächen für die beiden Darstellungsformen und der programmtechnischen Umsetzung (2MM).

Das Modul Lernwegauswahl beinhaltet

- die Konzeption von Abfragen, Lernwegen und Gliederung anhand lernpsychologischer und wissenschaftstheoretischer Kriterien und deren Implementierung (2MM)

Der Lern- und Lehrplaner: Evaluation umfaßt

- die Entwicklung eines experimentellen Designs inkl. Fragebögen (1MM),
- die Durchführung der Experimente mit Datenerhebung in Einzel- bzw. Gruppensitzungen (2MM),
- die Auswertung und Dokumentation (2MM),
- und die direkte Umsetzung in eine funktionale Benutzeroberfläche zur Planung des Lernweges sowie zur Organisation von Lernendengruppen (2MM).

#### 2.2.2. Entwicklungsumgebung (ges. 8MM)

Die Entwicklung des Simulationswerkzeuges umfaßt

- das Design einer graphischen Benutzeroberfläche (1MM)
- und die Implementierung der Routinen, über die die Simulations- und Visualisierungswerkzeuge mit der Maus manipuliert und frei miteinander kombiniert werden können (3MM).

Die Entwicklung des Skripters umfaßt

- die Entwicklung einer JAVA-basierten Skriptsprache zur Positionierung und Organsiation der Medien (2MM)
- sowie das Design und die Implementierung der graphischen Benutzeroberfläche mit Auswahlfenstern und Speichern-Laden-Funktionen (2MM).

### 2.2.3. Online-Umgebung (ges. 12MM)

- Die Aufgabe bei der Realisierung der Kommunikationsschnittstelle ist, das TCP/IP-Protokoll aus der RUBIN-GUI heraus auf verschiedenen Plattformen und in verschiedenen Netzwerkarchitekturen nutzen zu können (1MM).

Die Bereitstellung von Kommunikationsstrukturen für Lernende umfaßt

- die Implementierung eines Chat-Moduls (1MM).

Die Erstellung der WBT-Erweiterung für den Lern- und Lehrplaner umfaßt

- das Design und die Implementierung von graphischen Benutzeroberflächen, mit denen die Funktionen des Lern- und Lehrplaners auch 'online' genutzt werden können (1MM).

Die Entwicklung des Datenbankmoduls umfaßt

- die Entwicklung der Datenbankstruktur (2MM),
- die Integration der Medien (1MM)
- und die Bereitstellung von Such- und Editierfunktionen für den RUBIN-Client (1MM).

Ferner muß ein eigener Server aufgebaut werden,

- auf dem zur technischen Qualitätssicherung exemplarisch alle Module der Online-Umgebung installiert und getestet werden (2MM),
- und die WWW-Seiten und Serviceleistungen (Updates, News, Links etc.) bereitgestellt werden (2MM).

Jede Ausbaustufe des zweiten Produktionsabschnitts bildet ein Programmpaket, das auf einer lokalen Festplatte zu einer bereits vorhanden RUBIN-Version installiert werden kann (CD-ROM oder Internet-Distribution).

## 3. Kooperationen

Im folgenden wird aufgeführt, welche Einrichtungen bei der Realisierung des RUBIN-Projekts beteiligt waren und sein werden, bzw. welche Kooperationen in der Folgezeit notwendig sind, um die Produktionsabschnitte professionell abwickeln zu können.

### 3.1. Beteiligte Einrichtungen und Personen

Das Projekt wurde an der Universität Bielefeld von den im folgenden aufgezählten Einrichtungen initiiert und wird vom Universitätsverbund MultiMedia des Landes NRW über den Zeitraum vom 1.7.1998 bis zum 31.12.1999 gefördert.

Universität	Prof. Dr. M. Egelhaaf, Lehrstuhl für Neurobiologie,
Bielefeld	Fakultät für Biologie (federführend)
	Prof. Dr. H. Cruse, Abt. für Biologische Kybernetik/Theor. Biologie,
	Fakultät für Biologie
	Prof. Dr. H. Ritter, Lehrstuhl für Neuroinformatik,
	Technische Fakultät
externe	Prof. Dr. H. Wagner, Lehrstuhl für Zoologie/Tierphysiologie,
Kooperations-	math.-nat.wiss. Fakultät, RWTH Aachen

partner für Prof. Dr. A. Büschges, berufen auf den Lehrstuhl für Tierphysiologie,  
Evaluation math.-nat.wiss. Fakultät, Universität Köln

Als wissenschaftliche Mitarbeiter waren beteiligt: Dipl. Inform. Sören Lorenz (1.7.1998-31.12.1999) und Dipl. Biol. Wolfram Horstmann (1.10.1998-31.12.1999). Als wissenschaftliche Hilfskraft war beteiligt Dipl. Inform. Markus Oesker (1.1.1999-31.12.1999). Darüber hinaus waren studentische Hilfskräfte beteiligt.

Neben der Kooperation, die sich durch die oben beschriebenen Initiatoren an der Universität Bielefeld und die externen Kooperationspartnern für Evaluation formiert hat, haben sich seit Projektbeginn weitere Kooperationen entwickelt.

- In einer Kooperation mit der Fachhochschule Design in Bielefeld (Prof. G. Fleischmann) haben Diplomandinnen des Studiengangs 'Visuelle Kommunikation' (Iris Jagow, Martina Massong) in enger Kooperation mit den RUBIN-Mitarbeitern ein einheitliches Erscheinungsbild für das Gesamtprojekt RUBIN (Corporate Design) und die graphische Benutzeroberfläche für die Lehrsimulationen und das Simulationslehrsystem (Screen Design) entwickelt.
- Außerdem wurde eine Kooperation mit einem, von Absolventen der Technischen Fakultät und der Bielefelder Fachhochschule gegründeten, Software-Unternehmen ('media engineering') durchgeführt. In dem Unternehmen wurde eine Java-Multimedia-Bibliothek (JMB) entwickelt, mit deren Hilfe Oberflächen in Java implementiert werden können, wie es sonst nur von Autorensystemen bekannt ist. Die JMB unterstützt die Oberflächen-Entwicklung der RUBIN-Module. Da die JMB in RUBIN erstmals außerhalb firmeneigener Projekte Verwendung findet, wird sie von den RUBIN-Entwicklern einem Beta-Test unterzogen und gegebenenfalls erweitert.

### 3.2. Geplante Kooperationen

Um in der Folgezeit eine professionelle Abwicklung des Projekts zu gewährleisten, wären weitere Kooperationen mit Einrichtungen der Universität Bielefeld oder anderer Universitäten hilfreich. Für die Bewertung des Drehbuchs sollen – außer den bereits jetzt beteiligten Einrichtungen – auch Einrichtungen kognitionswissenschaftlicher Disziplinen einbezogen werden. Bei der Medienproduktion (v.a. Filme und Audio) kann das Audiovisuelle Zentrum (AVZ) der Universität Bielefeld helfen. Zur Erhaltung der hohen graphischen Standards wird die Kooperation mit dem Fachbereich Design der Fachhochschule Bielefeld aufrechterhalten. Für die Evaluation der didaktischen Prinzipien und lernpsychologischen Annahmen der Lehrsimulationen (formative Evaluation) und für eine umfassende Bewertung des Programms (summative Evaluation) ist die Kooperation mit einer Einrichtung für Lernpsychologie/Pädagogischer Psychologie und/oder Medienpädagogik geplant.

Bei der programmtechnischen Realisierung ist eine Kooperation mit der AG Praktische Informatik der Universität Bielefeld geplant. Dort entwickelte Multimedia- und CBT-Werkzeuge (multimediales Glossar, automatische Lernzielkontrolle etc.) können als Vorlage bei der Erstellung der entsprechenden Module in RUBIN dienen.





# RUBIN

Rechnergestützter Unterricht zur  
Biologischen Informationsverarbeitung in neuronalen Netzwerken

Fokus	Neurowissenschaftliche Modellsimulation.
Inhalt	Von Neurobiologie über Neuroinformatik zu Kognitionswissenschaften.
Zielgruppe	Lernende und Lehrende. Für Grund- und Hauptstudium, aber auch gymnasiale Oberstufe.
Basisversion	»Simulationslehrsystem« als JAVA-Applikation. Lokale Installation (Win32, MacOS, Unix-Derivate).
Ausbaustufen	»Lern-, Entwicklungs- und Online-Umgebung«. JAVA-Erweiterungen, auch als Serverversion.
Einsatzfeld	Selbststudium und Lehrsupplement. Mit Ausbaustufen auch für (verteilte) Gruppen.
In Kürze	RUBIN ist ein multimediales und interaktives Lern- und Lehrsystem im Bereich der Neurowissenschaften, in dem Lernende in didaktisch aufbereiteten Lehrsimulationen erfahren können, wie Informationen im Gehirn verarbeitet werden. Um dem stark interdisziplinären Charakter der Thematik gerecht zu werden, ist das Ziel des Projekts, einen gemeinsamen Rahmen für die Lehre in Biologie, Informatik und Kognitionswissenschaftlichen Disziplinen vorzuschlagen. Als vollständig in JAVA realisiertes Lern- und Lehrsystem mit einer graphischen Benutzeroberfläche, aus der heraus alle bereitgestellten Module erreichbar sind, ist RUBIN prinzipiell auf jeder Rechner- und Netzwerkarchitektur lauffähig und eignet sich damit für das Selbststudium am Heimrechner, zum netzbasierten Lernen in Gruppen an universitären Servern sowie für punktuelle Präsentationen in mit Multimediatechnik ausgestatteten Veranstaltungsräumen. Die modular angelegte Struktur von RUBIN erlaubt die Anbindung vieler weiterer Funktionseinheiten, die das Lernen und Lehren effektiver und komfortabler machen und RUBIN letztendlich zu einem Lern- und Lehrsystem mit Modellcharakter machen können.

# RUBIN in Modulen

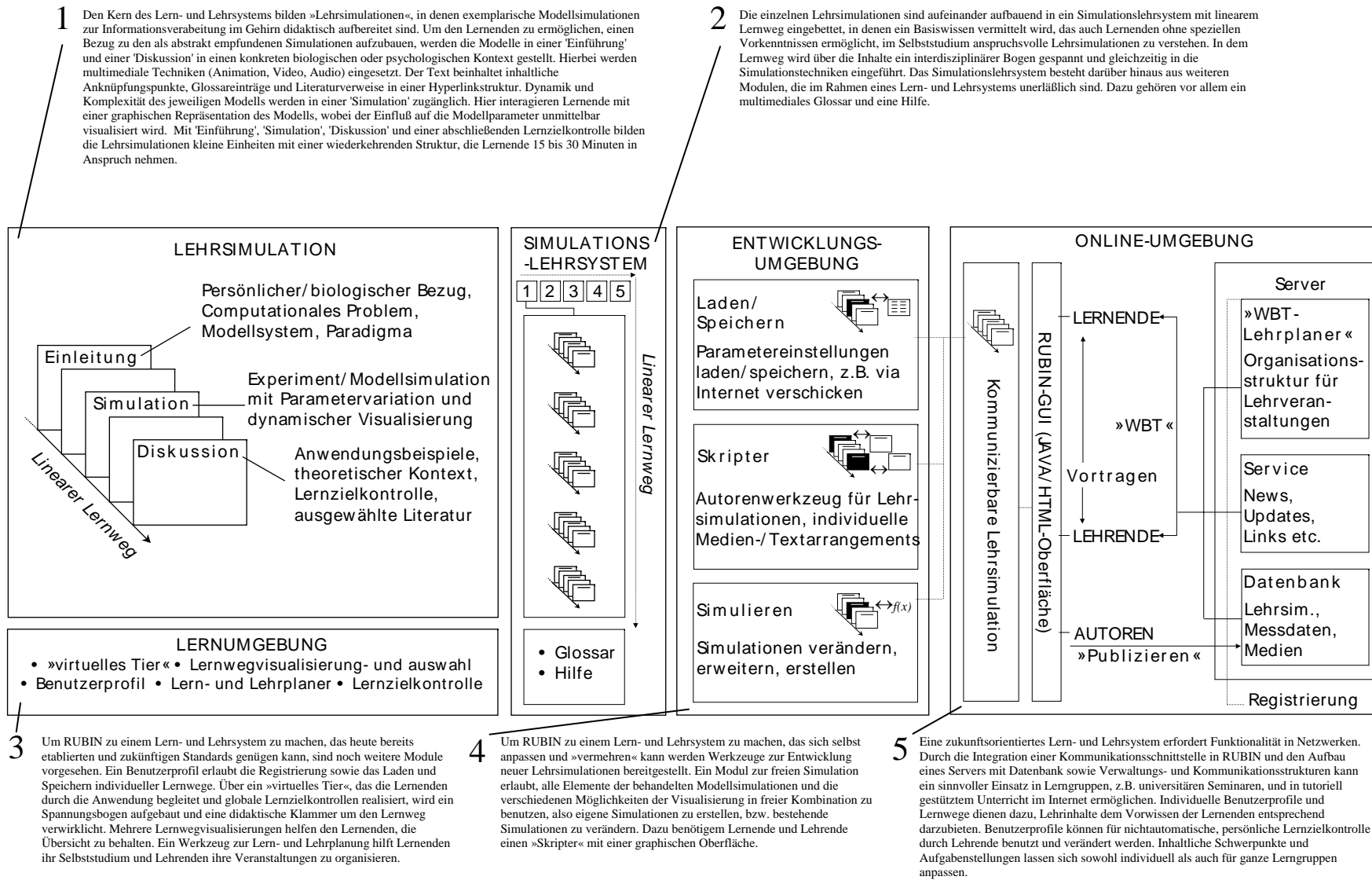


Abb.17 Übersichtsschema des RUBIN-Konzepts. Alle Ausbaustufen sind berücksichtigt.

