S.Tolkachev

NEURAL PROGRAMMING,

INFORMATION AMPLIFICATION

AND

INTERACTIVE SEARCH

(Translated from Russian by PromtX version 6.5.0.647)

St. Petersburg - Minneapolis
2007

UDC 004.032.26
ББК 32.973-01
T52
ISBN 5-7931-0450-X
T52

Neural Programming, Information Amplification and Interactive Search

In the book we discussed methods for programming consumer-oriented interactive Web-based applications with introduction of neural linguistic modeling and information amplification. Means of developing linguistic neurons and their implementations to solve various application problems are described. New approach to "raise" programs by using biological analogies has been proposed. Also the method to measure the interactive performance was introduced, which helps developer to build the optimal interactive search interface between visitor and internal database.

The set of programs allowing the reader to experiment with the elementary neural models are included in the book. Readers can download these programs and other additional materials from the book's web site.

The book is destined to a wide audience of programmers, web-designers, students and scientists having the basic skills of programming and who are interesting in developing the next generation of interactive Web.

Dedicated to my wife.

... There have been cases when unrestrained
fermentation of minds eventually led to the emergence of truths.

I. M. Sechenov "Reflexes of the brain".

INTRODUCTION

The ability to listen and understand the interlocutor is one of the most valuable advantages of the human dialogue. Asking questions or simply exchanging phrases during the conversation, we aspire to receive the answers adequate to our internal expectation. We equally are not satisfied with the messages which are not bringing us enough information, as well as with the answers in which its quantity considerably surpasses our internal restrictions established for the particular conversation. The meaning content in the answer or *the adequate reaction* of the interlocutor is one of the main criteria, defining the quality of conversation and influencing on change of internal state of the person during the dialogue.

Assuming, that the subject of conversation may have some target function which defines expected amount of information (desirable result); and there is amount of information in the each received messages, we may represent both of them in the form of some trajectories. Depending on the character of conversation these trajectories can take the diversified forms (Fig. 1). Sometimes the necessary result can be reached by the optimal way (a). In other cases, asking this or that question, instead of the useful answer we may get a plenty of surplus information which can take us far from the initial purpose of the conversation (b) and (c).

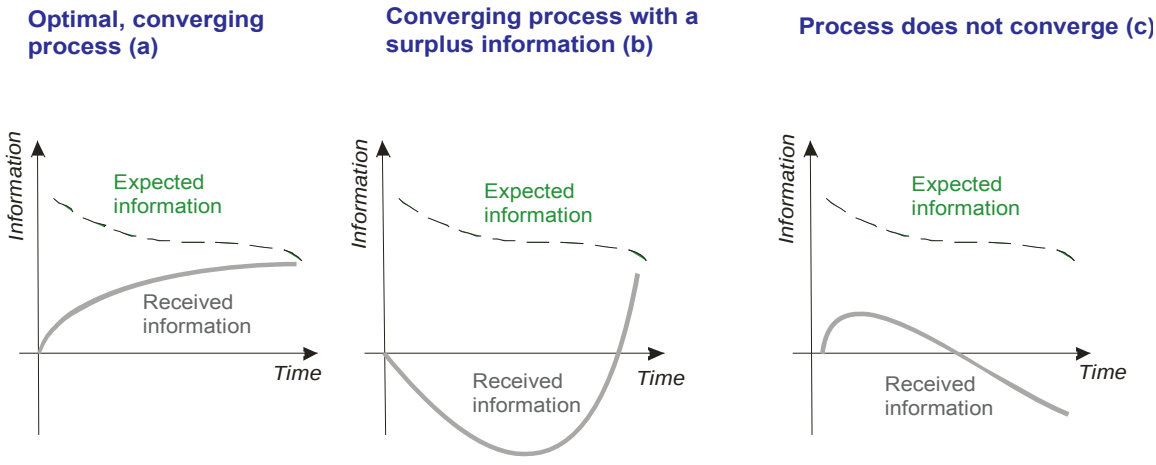


Fig. 1 Trajectories of processes of interaction

The stream of the words coming to the participant of conversation during the exchange by messages causes perturbation of consciousness which can be imagined as disturbances of a surface of water as a result of falling drops of a rain. Diffusion of information waves in consciousness, their interference with each other and interaction with deep internal processes in mind forms the complex system. Research and modeling of such systems can be conducted with

methods and means, similar to which are applied in the physics, describing the behavior of electromagnetic fields and particles, based on the wave-corpuscle dualism properties of a matter.

If to admit, that asking question **Q** we expect to receive the certain amount of the information **I(Q)**, and considering this, the interlocutor should respect this by providing the answer with amount of information equal to **I(A)**, then we can formulate the principle of information adequacy as follows:

$$\mathbf{I\,(A)\ \ \rightarrow\ \ I\,(Q)}$$

Where:

I (A) – the quantity of information in the answer **A** tend to
I (Q) – the expected quantity of information in the reply to the question **Q**.

In the Hardware Engineering, information measurement of characteristics for various kinds of devices such as: memory, processors, communication channels, etc. are widely used. And in the Theory of Information, there are well known ways to measure amount of information of streams of messages. It will benefit to all if we could use similar measures for the estimation of the quality of interaction and the status of systems participating in this process. However the majority of information characteristics in Hardware engineering are static, in comparison to a human dialogue which has extremely dynamic character.

Models of such processes, definition of their information characteristics, a finding of optimum trajectories of interaction, etc. belongs to the group of the most complex and labor-consuming tasks in modern programming. Among the major factors defining their complexity first of all we need to consider the following:

- A big number of dynamic parameters;
- Permanent adaptation and development of internal structure and functions;
- Absence of precise criteria of quality of behavior.

Attempts to solve these tasks using traditional programming only, leads to the problems which have roots lying in the logical basis of algorithmic modeling. Programming of algorithms, based on the interpretation of sequences of passive logic constructs, in many cases allows achieving acceptable results, but *the program* and *result* essentially differ and are obviously separated from each other.

In neural models the active elements possessing internal dynamic properties are used and they are capable to *accept and transfer* signals independently. Thus, executive elements and results represent a single whole - dynamic environment, the space of parallel processes constantly changing neuron's conditions. *And one of ways to create such models is the **Neural Programming**, which we shall define as methods for organization and management of active elements in the interpreting systems*.

Neural Programming possesses one indisputable advantage - the nature during evolution has already constructed huge quantity of biological prototypes, wide spectrum of which includes as nervous systems of the elementary multi-cellular, as well as a human brain, most complex of the organized systems known on today, therefore the knowledge of biology and neurophysiology can provide the essential help at programming artificial neural systems.

The wide scattering in complexity of modeled systems assumes to employ appropriate programming technologies and methods. The same as much as in algorithmic programming methods the construction of big Operating System differ from methods of writing small private programs, in neural programming, the process of construction of big models essentially differs from the creation of small systems.

Technologies of construction of information systems are concerning to the area of special interests in programming. Among many works in this direction it is important to mention E. Dijkstra's early article "*The Structure of 'THE' Multiprogramming System*" [7], in which he has formulated principles of construction of the programming systems, essentially differing from widely accepted then technologies of programming (see for example, F. Brooks "*Mythical man-month*" [3]). In this work Dijkstra not only considers the general technological principles for software development, but also he'd introduced synchronizing primitives - semaphores by means of which he managed to solve the asynchronous problem in parallel processes' programming.

The new technology and new structural-functional design in his work are highly interconnected, and this aggregation allows him to achieve solutions which were more progressive and reliable in comparison with known then systems.

Dijkstra's ideas about programming, his elegant theoretical and practical solutions continue to remain attractive to everyone who is connected with this discipline. In one of his last interviews "*Discipline of Thought*", to Dutch broadcasting company *VPRO* in 2001, arguing on an essence of programming, he speaks the following:

> "There are very different programming styles. I tend to see them as Mozart versus Beethoven. When Mozart started to write, the composition was finished. He wrote the manuscripts in one go. In beautiful handwriting too. Beethoven was a doubter and a struggler, who started writing before he finished the composition and then glued corrections onto the page… In order to compose, you have to write scores. But to be a composer is not to write scores. To be a composer is to conceive music. In early days of programming, you have to write machine code. Meaningless sequences of capitals and numbers. That's the analogue of writing scores. People thought that *that* was programming. Later that was made easier by the invention of the higher programming languages: FORTRAN, Pascal, C++ and suchlike. People thought that those languages would solve the programming problem. But when you looked closely the trivial aspects of programming had been automated while the hard ones remained…"
>
> http://www.cs.utexas.edu/users/EWD/videos/NoorderlichtVideo.html

Dijkstra touches upon an especial theme in the programming, connected with individual creativity and he uses interesting analogies to music. To continue this analogy, it is possible to compare interpreting systems which necessary for construction of neural models with various musical instruments.

To perform great and complex pieces of music in symphonic orchestras the diversified groups of tools are used. For chamber music, quite enough just one musical instrument such as piano. Among the variety of software tools intended for the individual (chamber) programming, we are especially interested in *HyperCard*. In this system, combination of multi dimensional layers with easy-to-use objects and dynamic interpretation of scripts makes it as a perfect environment-interface to many applications, including neural models. The architecture of *HyperCard* has served as the prototype for web-pages and even today it surpass functionality of dynamic programming *DHTML* in many respects. Unfortunately, in the 90th *HyperCard* has been frozen and now has practically stopped the existence. The reason for cancellation was that management

of *Apple Computer, Inc.* could not understand and estimate its strategic value. In the interview given in 2003 *CNET News*, John Scaly who was in 80th years the president of *Apple* has told about it:

> **CNET: Any missed opportunities that you wish you could do over?**
>
> **John Sculley:** As I look back on things that I wished we would have done differently when I was at Apple, I think one of the biggest missed opportunities, and it was on my watch, so I feel responsible and disappointed that we didn't do more with it, was HyperCard. It was created back in 1987 by Bill Atkinson, Apple's first software programmer. We could never figure out exactly what it was. We thought it was a prototyping tool. We thought it was a database tool. It was actually used by people as a front-end communications device for TCP/IP to connect the Internet to large Cray computers.
>
> We weren't insightful enough to recognize that what we had inside of HyperCard, essentially, was everything that later was developed so successfully by Tim Berners-Lee with HTTP (Hypertext Transfer Protocol) and HTML (Hypertext Markup Language). We didn't call it that. But essentially, we had all that hypertext, radio buttons and linking capability architected in the original HyperCard. In hindsight, I wish Apple had recognized that we had a huge opportunity to go take our user interface culture, and our know-how, and applied it to the Internet. I think we would have had a very different story for Apple during the 1990s. But that, of course, is hindsight.
>
> *http://news.com.com/2008-7351-5085423.html*

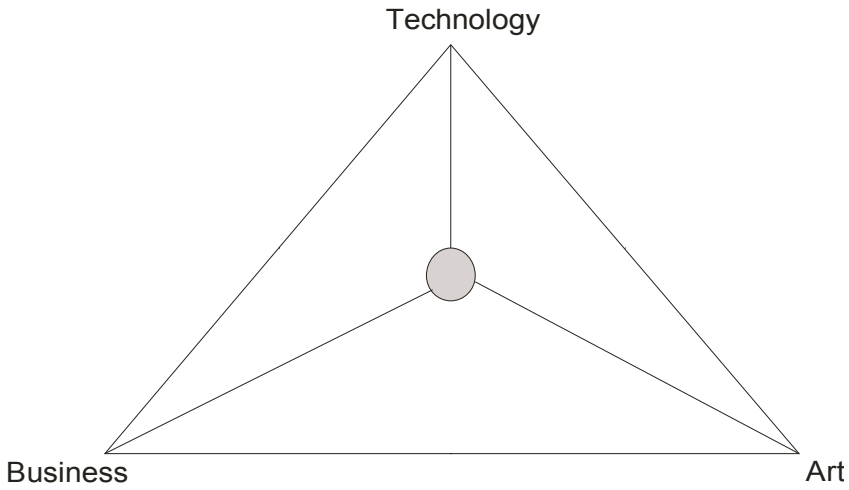
In this sense between *HyperCard* and neural networks is much in common. In the end of 60th Neural Networks have been declared as unpromising, and it was required more than two decades, until the interest to this domain has revived again. However till now this interest has more mathematical, than programming character. Modern researchers in the field of neural networks first of all explores mathematical problems of pattern recognition and classification which require finding efficient methods for establishing and adjusting weights of links between neurons. On the other hand, programmers should be interested in learning dynamic properties of neurons, their unique abilities to connection, reconfiguration and reproduction. These properties of neural networks surprisingly coincide with properties of the Internet which is the spontaneous, decentralized system and in which there is a huge quantity of parallel processes.

The Internet itself was founded as a system for exchange of messages which can be presented in the diversified forms, and that allows to consider it as greater interactive system. Thus interaction can occur either with participation of the person, and by means of various bots - the artificial representatives, capable automatically to look through contents a web-pages, to answer questions of visitors, to do rates on auctions, etc. For construction such bots are used various methods of programming; including neural networks, to which the increasing attention is paid recently.

Result of programming, either neural or algorithmic, finally is the software product possessing certain commercial properties. In the process of constant increase of number of users in the Internet, the needs for the advanced systems, capable to simplify processes of dialogue between the person and the computer, increases proportionally. It is possible to have mixed opinion considering Bill Gates' "*An Open Letter to Hobbyists*" in which he calls for commercial professionalism in programming (all programs which are presented in this book, belong to category of *Open Software* and can be used freely), however that fact, that in the market to neural systems today is shown heightened interest, allows to hope, that besides professional curiosity, neural programming will manage to involve commercial interest of developers.

The destiny of software product depends on set various and often inconsistent factors. Creative aspirations and searches of interesting decisions collide with commercial requirements and technological restrictions. If to present the process of real programming, as a process of

permanent finding of optimal elastic balance between points in the space with the set of constraints:



then the Art is one of its most important components. Art plays a fundamental role in finding a balance between practical sense and theoretical restrictions just because it helps to establish compromises and to connect contradictory and conflicting approaches [21].

The aspiration to such connection, search of various approaches, methods and models which will allow solving broader set of problems constantly arising before programmers, is an overall objective of this book.

ENERGY, INFORMATION AND KNOWLEDGE

Both theories of processes of transformation of energy and information are founded on the common and general principles, which allow assuming, that practical designs of energy-converting machinery and computer systems should possess some similar properties. Being based on these analogies, we shall try to apply such physical concepts as *work*, *capacity*, *efficiency*, *a principle of uncertainty* and others to the analysis of information systems. Also we hope that the experience of designing and management of mechanical and electrical mechanisms accumulated in the industrial world will help with designing information machines.

With growth and maturity, the information technologies gave birth to such kinds of activity as virtual enterprises, electronic trade, e-learning, remote diagnostics, search service, etc. which have opened to mankind essentially new opportunities, but simultaneously they also created a number of new practical and theoretical problems. Among them - extraction of knowledge from large distributed sources of data, removal of pollution from streams of messages, natural languages processing, integration of diverse communication channels, and others which have in turn caused to revision and expansions of some of methods and technologies of modern programming. For example, the increase in accessible volumes of data in many cases does not lead to the positive result, but contrary, to causes of the overflows in communication channels and deterioration of decision-making. Attempts of creation of the unified interfaces and invariant and independent, in relation to the particular person, forms of data presentation, are rejected by users owing to natural aspiration of people to individualism. Navigational menus and variety of search helpers, instead of reducing a trajectory of viewing, considerably increases the time of search for useful information. The same inquiry sent into the search service from two computers by two different people - will return the same results, without dependence of what they meant. Before the Internet, problems of data processing basically have been connected with search and reconstruction of the missing information from the limited local volumes of data. Now, we have new urgent problem - individual filtration and transformation of the information receiving from constantly increasing and practically unlimited streams of the messages coming from the outside. Nowadays, the static forms of presentation used in traditional theoretical programming based on unequivocal logic expressions, mismatch the realities of the Internet where programs, structures and data are under the constant dynamic transformation.

Among the all set of practical problems which being solved in computer science during the long period of its existence – data entry may be referred as an “eternal” category. Just studying data entry it is possible to track the evolution of several generations of hardware, software and data structures, which has led today to appearance of the centers for remote service - *Call Centers*. In 2005 in the world there was more than 70 000 Call Centers in which worked more than 3.5 million people. Taking into consideration, that one of the main criteria of successful business is constant improvement of quality of service for users, becomes obvious, what strategic value associated with these centers in the modern business world.

From the very first moment with the advent of punched tapes and punched cards, structure of data and technologies of their processing were gradually transformed from the slow in time batch processing of preparation and accumulation of data, to the real time systems, which reacted to the inquiries immediately. The modern Call Centers are complex switching-dispatching systems, in many cases territorially distributed worldwide. Among of key functions of such centers are - accept the message, define its sense and connect the subscriber to corresponding service which

could be another person or web page with a proper content. Today we already have the situation when the quantity of the people initiating various inquiries or the messages by phone, in the form of e-mail or in a chat, already considerably surpasses the available personnel, and it is obvious, that in process of globalization of the international community this problem will be aggravated more and more. Creation of the virtual representatives (agents) capable in a certain measure to understand entrance inquiries and adequately to react to them, whenever possible answering or switching on corresponding services, is one of possible solutions of this problem (Fig. 2). To make this solution effective is necessary, that technologies of training of virtual agents were simple and available to managers who in many cases are not professional programmers.

The quantity of bits of the information circulating in the Internet is comparable to numerical characteristics of physical objects on micro- and macro-levels. In physics, transitions from one level of conception to another require the new methods of research, which could be essentially different. Statistical thermodynamics, mechanics, and the molecular physics are cardinally varied, but at the same time they are well coordinated sections of one science. Unlike physics, computer science has no universal theoretical bases so far, within the limits of which representations and methods of the analysis are structurally united similar to corresponding sections in physics. The classical theory of information introduced by Shannon in the forties and based on the analysis of sequences of symbols, transmitted from the sources into the receiver, and used the Boolean logic which is dealing with exact binary values, has been ceasing to work when it is dealing with data flows which on many orders surpassing the capacity of the receiver. In physics phenomenological and analytical methods traditionally developed in a way, allowing to scale the explanation from the high professional theoretical level to the simplified form accessible by general public. For example such ratio as dependence between temperature **t**, pressure **P** and volume **V**, in thermodynamics, or concepts of *efficiency, work, capacity and friction*, well-known from the school program, with sufficient accuracy and a degree of mutual understanding coordinate representations of consumers, engineers and scientists. Such theoretical and practical scaling of knowledge - one of essential conditions for successful integration of scientific researches together with designing, manufacturing and utilization of power and information machines and systems.

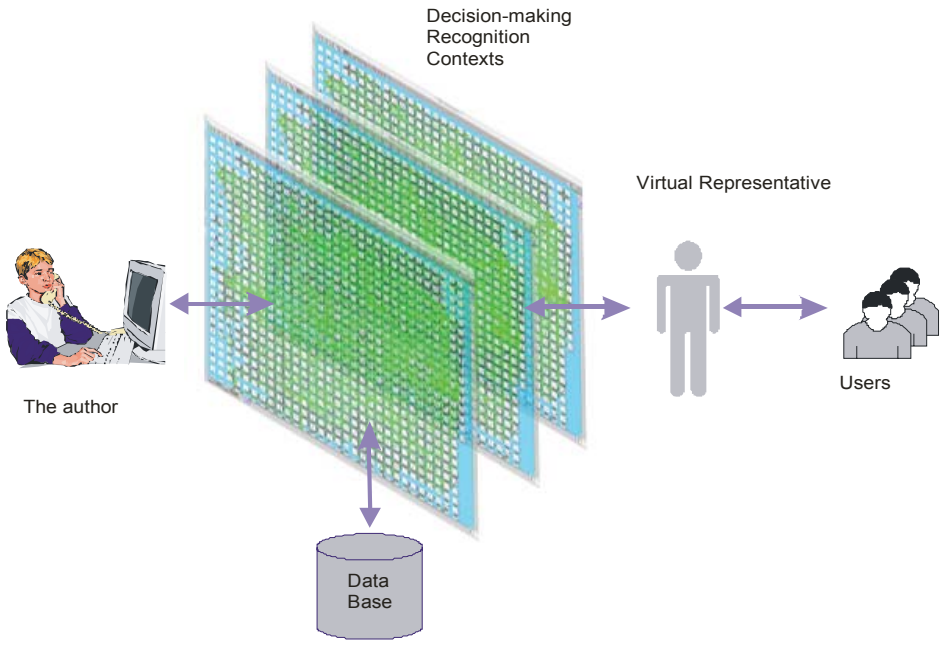
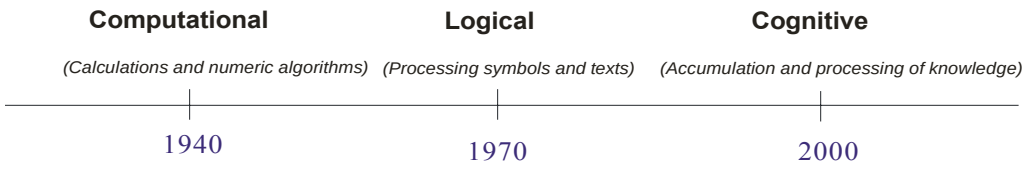


Fig. 2 The virtual representative

We may dare to presume, that ideas of information amplification and neural programming for the first time have been formulated in 1945 when there were two works, in many respects predetermined development of computer science for decades forward: von Neumann's report - "*First Draft of a Report on the EDVAC*" [20] and Vannevar Bush's article in *The Atlantic Monthly* - "*As We May Think*" [4]. Formal models of elements and structures of automatic digital computers are mutually complemented together with sketches and prospect of the future of information systems. The function structure of von Neumann's computer and hypertext model of knowledge of Bush, have appeared during that moment when abstract reasoning on the nature of calculations, logic of thinking and knowledge have started to find a practical embodiment in the form of real information systems. At this time the latent stage in history of computer comes to the end and consecutive development of computer's architectures, which for today can be divided into three conditional periods, has began:



Mathematical calculations and computing algorithms dominated over the theory and practice of computer facilities up to the middle of seventies, when logical and functional programming intended for processing of symbolical and textual data acquired the importance. In the middle of nineties, with the advent of the *Web*, the processing of knowledge emerged: management of distributed streams of messages, linguistic analysis and synthesis, recognition of images, integration of various forms and ways of dialogue (phones, radio and TV) together with computers.

Respective changes have occurred at the same time in methods of programming - for example, the languages focused on processing of structured data and streams of messages such as *C++* and *Java*, essentially differ from the first generation of languages for numerical and algorithmic calculations, such as *FORTRAN* and *ALGOL*. The main difference between them consists in a degree of integration of program elements and association of functions of processing together with structures of data that has led to occurrence of *objects* - to qualitatively new category in programming languages. We may suppose that further development in knowledge processing will lead to the deeper integration of program's *objects* together with other entities, first of all such as external structures of data and the distributed functions, and as consequence, to occurrence of new ways of management by ensembles of diverse systems.

Prototypes of such integration already can be observed now in modern Object or XML databases, in programming systems of Agents (*Agent Oriented Programming*) and, at last, in the widest kind, in dynamic properties of new generation of web-pages (Web-2, Web-3). It is conceptually possible to assume, that *Web 3.0* will possess much wider spectrum of dynamic properties which will be carried out with tools and environment such as *Ajax, DHTML, XMLHTTP* and other, capable to organize interaction of a plenty of the heterogeneous objects forming complex semantic networks. Biological systems in this sense are the good sample for comparison as they are perfectly adapted for interaction with a lot of complex diverse objects in an external world and now are sufficiently studied, that knowledge of their properties could be used by development of new methods of programming.

It may easy to considered, that to some extent, all problems of processing of the information related to knowledge processing. Application for knowledge accumulation and processing is an overall objective of all modern information technologies both in global sense, and in local realm. Individual knowledge of separate employees in a combination to corporate knowledge and data forms a powerful component of the whole system of values in the modern info - industrial world. So for example, leaving or moving of the worker of the enterprise should not be accompanied by loss of its individual experience which has been saved up during work on a certain place. One of ways of the decision of this problem is application of systems of training *(E-learning)*, capable to perceive knowledge of the expert and to transfer this knowledge to interested persons. Such systems are traditional in education - in 2003 the volume of their sales has made more than 2 billion dollars, however in last years they find more and more wide application in the industry and sphere of service. Dynamic and semantic properties of *Web 2.0* presume to organize process of accumulation and transfer of knowledge by natural evolutionary way, keeping thus the habitual "human" form of dialogue between the expert and users. However transformation of data into knowledge the same as transformation of energy into the useful work will demand special mechanisms, and as the basis for them we shall use the neural models.

AMPLIFIERS, NEURONS AND THE INTERNET

In his article "*As We May Think*" [4], arguing about the subsequent generations of systems which can raise efficiency of research work of the person, Bush very figuratively writes about essentially new forms of encyclopedias which being connected to the personal device - *memex*, can amplifying complexes of the associative-connected knowledge and those traces which in this knowledge the researcher can leave. Following Bush, it is possible to tell, that information amplification is an extraction of hypertext data from the dynamic distributed networks and their transformation during interactive communication with the person. Bush was apparently the first who has accurately enough defined the basic components of the intellectual amplifier: associative-connected dynamic data, the personal device-converter, and user interface.

To better understand principles of the information amplifier, we shall look at the process of amplification in the simple transistor. If to neglect technical details related with its internal technical structure, the transistor is the converter connected to an energy source, receiving the weak input signal and producing the output signal gained on power (Fig. 3).

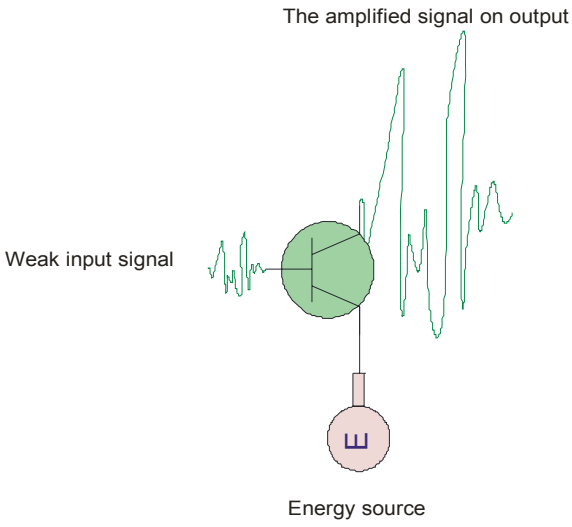


Fig. 3 The amplifier of electric signals on the transistor

Right at the beginning of his article, Bush emphasizes, that the economy is a factor which transforms ideas into a reality. Neither the talent of Leibnitz, which in 1673 has managed to create the calculation device close under the characteristics to a modern calculator, neither all resources of the Pharaoh, even if he possess all the knowledge of modern technologies, could not transform idea into practically working product, if the cost of development and operation will not be within the limits of economic feasibility [4].

Six decades, since the time of occurrence of Bush's article, was required, till the information alongside with energy has appeared as economically accessible resources for a greater part of mankind. And this resource can be practically freely used by any person on the Earth through the Internet - a huge, dynamical and constantly extending infrastructure of data (Fig. 4) which absorbs, stores, and reflects practically everything, that occurs in the Earth nature or is created by the person.

The majority of data in the first generation of the Internet is presented in the form of the pages and created in the manner as if they have been intended only for reading by the person though physically people capable to read only very smallest part of what can be received from millions web-servers.

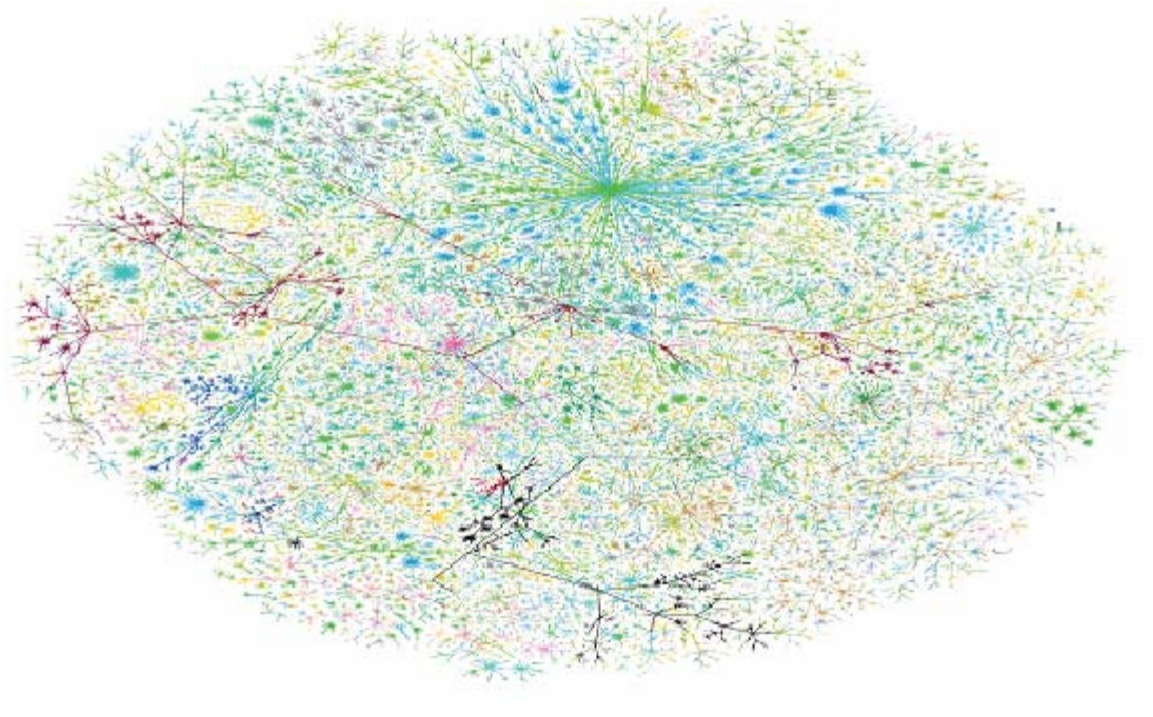


Fig. 4 Picture of the Internet at 01.01.2000
Copyright (c) 2001 Peacock Maps, Inc. and Lumeta Corporation
(http: // www.peacockmaps.com)

During the rather short period which has been since the moment of occurrence of the first static pages in format HTML, the Web promptly evolve in a direction of increase of functionality of pages and expansions of ways of their transfer.

J. Martin has noticed, that when in our world have appeared the essentially new technological achievements, for some time they are continue to be used by means of their predecessor [17]. For example, in TV, long enough time, since the moment of its occurrence, the most part of news was broadcasted with announcers reading the text, and the television image was only a background for audio-messages. And similar to that, in the first generation the Internet, pages, which actually duplicate printed materials, still continued to be used. However already today, integration of all basic channels for data transmission as well as combinations of various devices (computers, phones, radio and TV) already happened and developers has an opportunity to use new functionally-active objects, instead of passive pages of the text, for the organization of interaction with the human.

Hypertext references located inside of pages, which originally formed static networks of links are developing intensively, and quickly being transformed to dynamic formations into which they are integrated together with complex structures stored data (Fig. 5).

Pages has been transforming into active objects which in turn, are united in clusters, which can generate new structures and so on, that surprisingly reminds behavior of biological or physical groups of cooperating objects and also behaves quite according to Bush's ideas about traces which the person can leave in systems of knowledge.

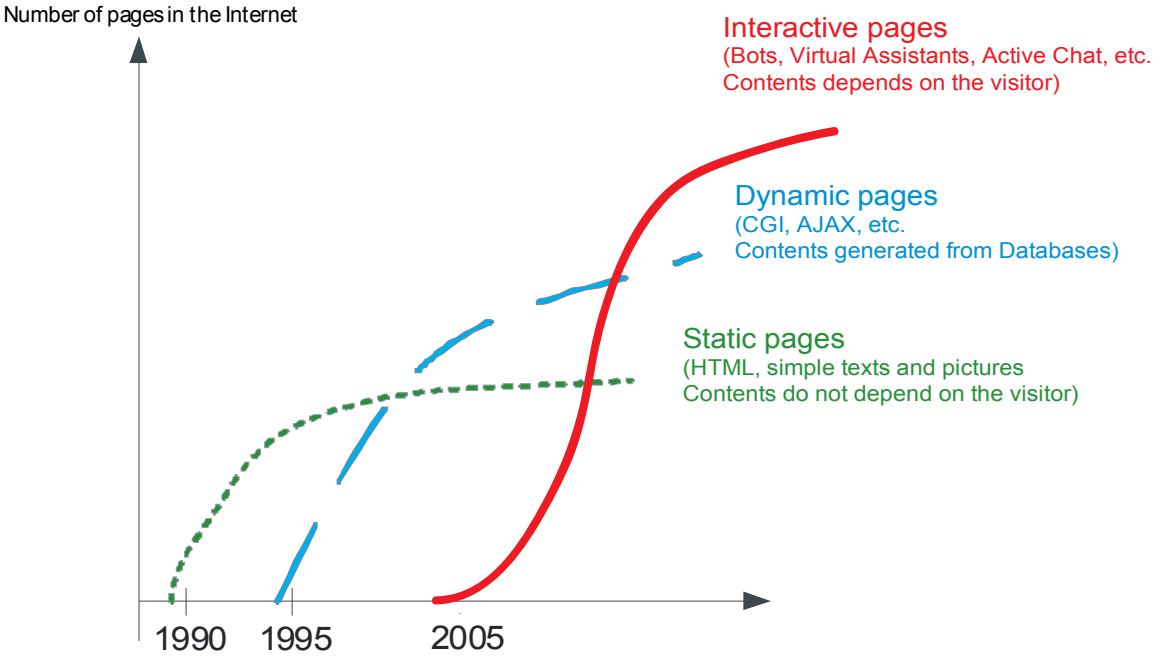


Fig. 5 Static, Dynamic and Interactive Web

In the report on the structure of the computer von Neumann uses neuron as the prototype for the basic computing elements of the automatic digital computing system and the model of biological nervous system underlies generic reasoning about computing architecture. Because the nature of the major problem which had being trying to solve by computer developers at that time was mostly calculations, von Neumann first of all considers arithmetic properties of neuron and uses it as a digital binary element which can carry out base mathematical functions.

The transistors which have appeared in 1948 possessed two steady conditions (Fig. 6). They appeared to be a quite successful decision for representation of binary data, and since then analogy between the transistor and neuron as binary devices, were for a long time fixed in the theory and practice of computer science.

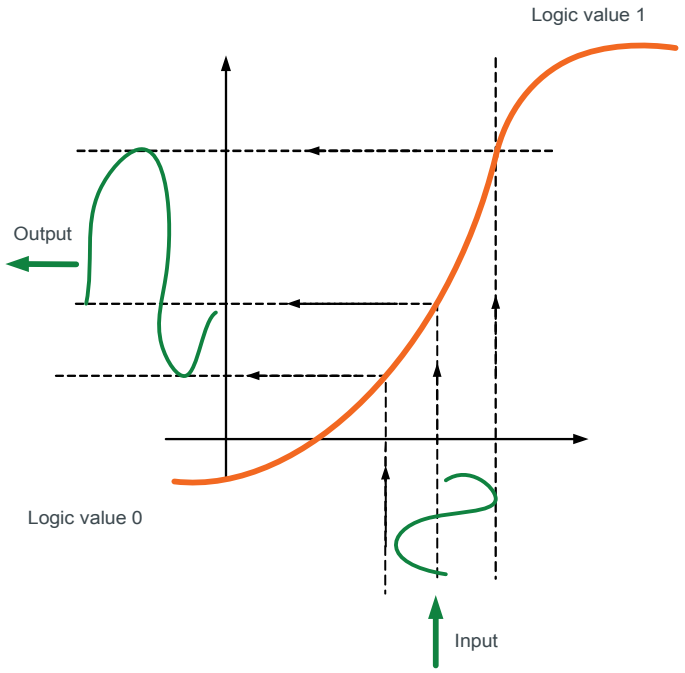


Fig. 6 Transistor's Current-Voltage characteristic

At the same time in other branches of electronics the different property of transistors based on transformation of a signal in the zone of transient between saturation regions is widely used.

Such transformation in transistors always has approximate character and possesses distortions which may be compensated in the certain degree, by using of various additional elements. The dualism of the transistor in which discrete behavior is combined with continuity, is a good example of a combination exact and approximate *(fuzzy)* properties in one device. Neuron also quite corresponds to this analogy; uniting the certain stability together with intermediate approximate conditions which it can hold depending on character of excitation and its functional purpose.

If to admit, that the information amplifier (Fig. 7) has structure, in a general view similar to structure of the amplifier of electric signals, then in such case the Internet can be used as a universal source of the information, input and output signals can be presented by means of audio and text messages of oral or written speech, and the converter can be realized in the form of some virtual machine - *Neural Virtual Machine (NVM)* which can be loaded into the personal computer or other device, similar to *Java Virtual Machine (JVM)*. Such information amplifier can be built in to the variety of systems.

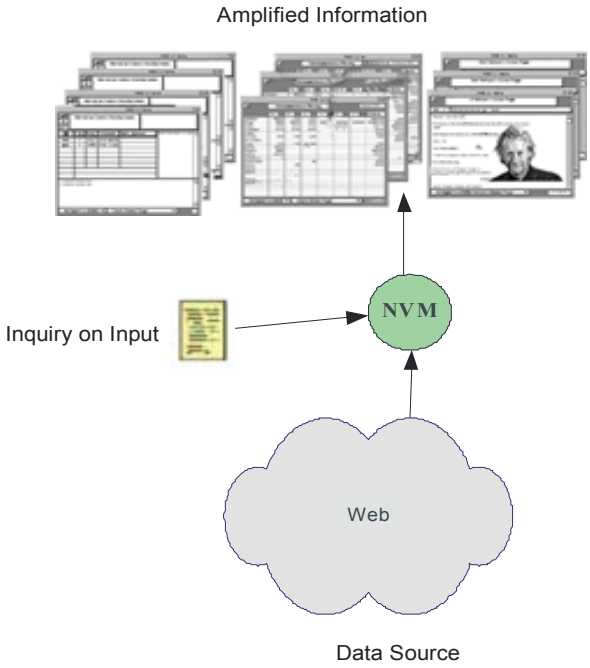


Fig. 7 The information amplifier

It is possible to assume, that success in the near future in nano-technology will allow implanting sensor devices directly in a human body, and in this case the form of dialogue will be significantly differ from speech. For this reason we hope, that neural models are that adaptive mechanism capable to natural integration with future symbiotic man-machine systems.

PROGRAMMING OF REACTIONS

Continuing search of analogies which can help us with designing information amplifier, we shall address to the work of I. M. Sechenov - "*Reflexes of the Brain*" [37]. Analyzing brain's behavior, Sechenov considers it as 'a black box' reacting to excitations of sensory neurons. He is making a simple but yet very far-reaching conclusion – "*all external manifestations of the functioning of the brain can be reduced to muscular movement*". Having this consideration, programmer may paraphrase it, using the scripts as an analogue for muscles, and the neural core as an analogue for brain – "*all external manifestation of Neural Virtual Machine can be reduced to external scripts execution*".

Let's imagine the Neural Virtual Machine (the information device and brain analogue) inside the environment of programming modules, each of them can be loaded and executed as a result of excitation of correspondent artificial motor neuron. In such case all external demonstration of internal processes of this machine can appeared as a result of call and execution of corresponding programming modules. Such a call can occur at the moment when excitation level of the motor neuron connected with this module through the output layer of information device, will exceed the threshold level.

If to admit, that with the programs distributed in the Internet can be associated a unique address and their own interpretation environment, then their executions may be initiated by sending request (for example *HTTP-request*) from one system to another (Fig. 8).

In this case the result of a neural layer's performance may be presented as execution of asynchronous parallel processes, and some of them may return data back to original system, and another will make changes in its own environment. Such a way of calls of program's modules allows essentially simplifying implementation of interfaces and reducing a problem to the answer to a question: how amplification and inhibition of signals can lead to adequate reaction when these signals associated with the words and meanings?

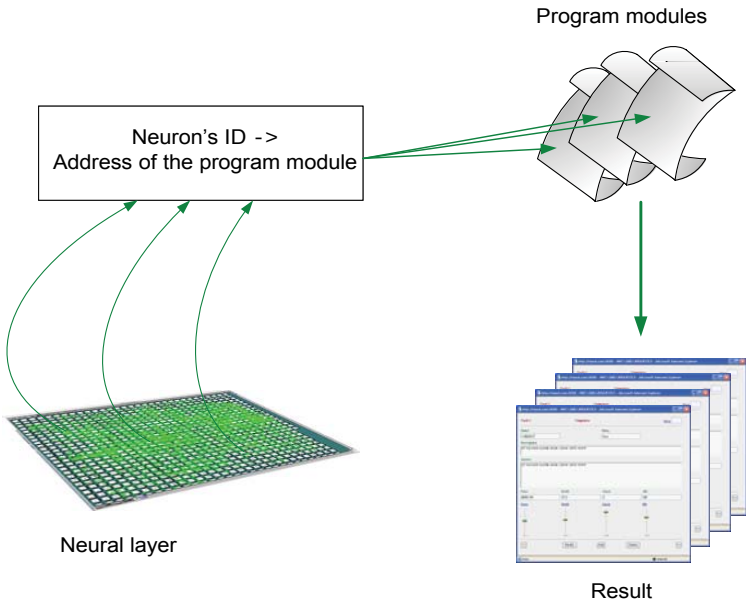


Fig. 8 Calls and execution of program modules in the neural environment

In physiology, the classification of all muscular movement into *involuntary* and *voluntary* is connected with practical impossibility to track all logic chains of consecutive actions which finally cause corresponding muscular contractions. At first sight, such classification has no meaning in programming where all streams of actions possess the certain determinism that allows to be assured in solution and estimations of accuracy of the results in each particular case. The situation varies significantly when we are observing hundreds millions of computers distributed in the Internet. In this case the programmer gets in a situation similar for the physiologist and the physicist when transition from one level of representation to another, requires applying of a different way of perception and research - exact knowledge of a code and a prediction of its behavior, does not have a sense in systems where changes of codes and data occur in such scales and with such speeds, that we cannot receive the determined picture all of their conditions.

Already mentioned *HyperCard* has been included by Apple as a component of *Mac OS* in 1987, long before appearance of *DHTML*. Its author - Bill Atkinson, one of leading programmers in Apple, has managed to construct the integrated system which during the very short time has attracted millions of users to the process which today we call as Web-programming.

The basis of *HyperCard* constitutes a set of working objects called *cards*. On the surface of the card it is possible to create (both manually, and automatically), variety of predetermined objects - buttons, text fields and images. The interpretable code (scripts) can be connected with each object. Executions and interpretation of scripts are carried out through the simple mechanism of events management. Cards are united by backgrounds, and organized in stacks. *HyperCard* works in a mode of direct interpretation and has unique and very easy system of identification and addressing of objects. The concept of *n-dimensional* programmable space in *HyperCard* is rather attractive owing to convenience and simplicity of access and execution of scripts. We shall use terminology and the architecture following to *HyperCard* because it has more user friendly architecture compared to modern *DHTM*. Insertion of the neural virtual machine in such *HyperCard*-like executive environment can be compared to connecting a brain to muscles and sense organs (Fig. 9).

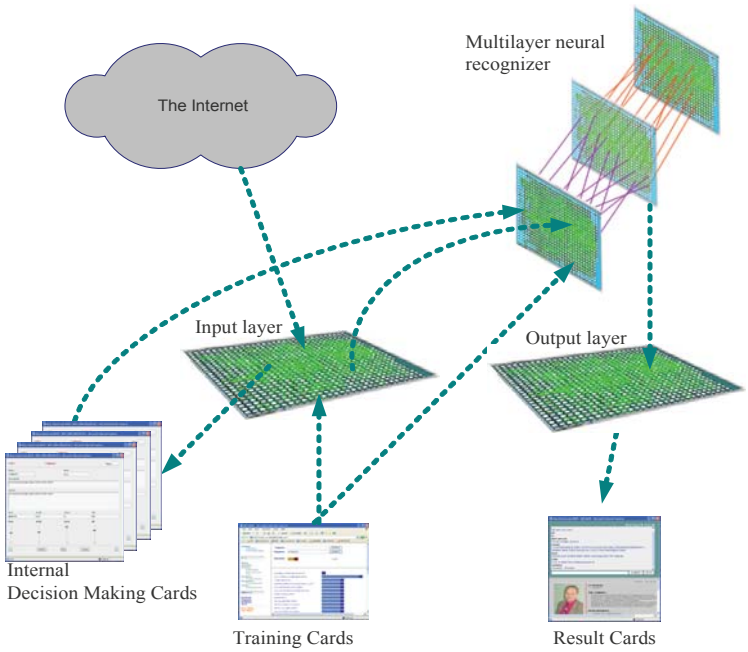
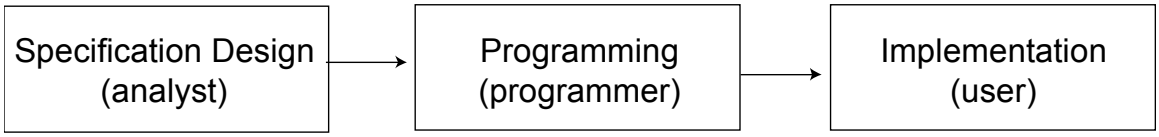


Fig. 9 Interaction between neural layers, cards and the Internet

As a matter of fact, according to this model, *Neural Programming can be defined as a process of creating neural objects and establishing connections between them and scripts, determining dynamic rules of their behavior for transferring excitations*. In cases when the number of objects and connections are relatively small, construction of neural models is a trivial problem. Difficulties begin when we need to connect a plenty of neurons through the big number of layers with the external mechanism in structure of which there can be many executive functions. The scale of a problem defines technology of its solution, and in programming there are many various methods of the analysis and synthesis of the complex systems, which are preceded to coding. Generally we can subdivide these methods into three major groups: flowcharts of data flows and managements, decision-making tables and structural-functional diagrams.

Use of preliminary analytical and specification methods allows considerably raise the productivity of all process of programming, first of all because of specialization and distribution of works on designing by the optimal way. Structurally functional modeling and other researches, based on decomposition and definition of functional properties of modules, such as *HIPO* developed in 70th years by IBM [24], have created a basis for visual models of very complicated systems. In modern software development the process of programming (coding), preceded by highly visualized design and followed by deep user involvement during implementation:



The technology of designing of interactive dialogue systems and their linguistic maintenance can be based on the same traditional methods of software development. At the first stage, the generic linguistic model is formed, which later will be coded in the form of neural structures. Neural structures are trained and corrected during the dialogues with the expert, however unlike traditional programs, they will continue to be trained and extended during the all period of its life, even after their implementation.

NEURAL PROGRAMMING ENVIRONMENT

To create and maintain neural models we will need an appropriate programming environment which will support neural-signals propagations, structural and functional design and development, and also will interpret responding reactions. The example of one of such possible environment configuration which united the chat, speech recognition, neural core, dynamic library of program modules, SQL database and other functional components shown on Fig. 10.

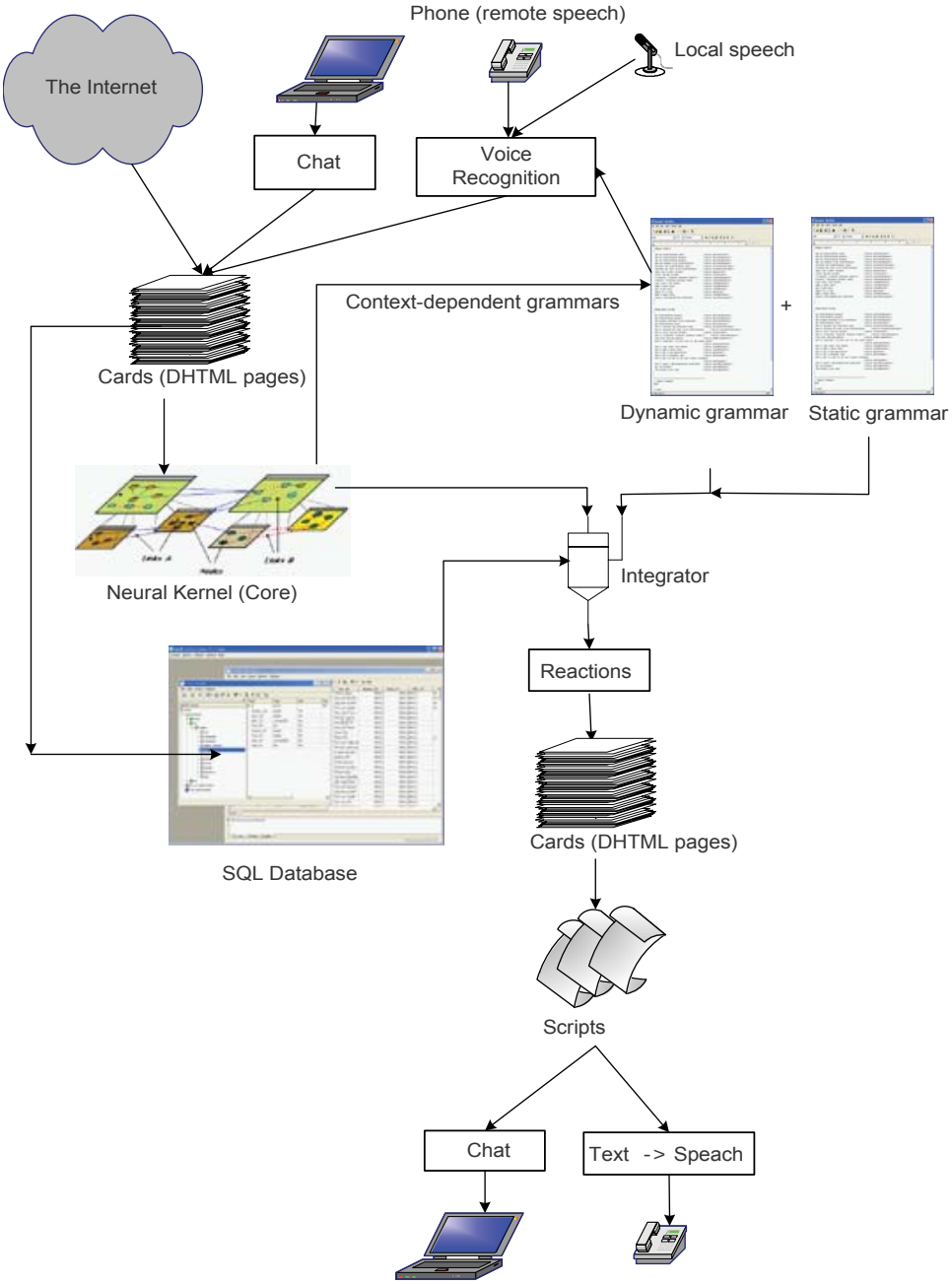


Fig. 10 Neural programming environment

The technical term - *programming environment* in our case defines programming languages, interpreters and compilers, protocols, technologies and tools. The *neural programming environment* which we shall use in our development includes:

- Interfaces to various speech and text messages input/output channels;
- A neural kernel (Core), which provide the support for artificial neurons together with their links (nerves), capable to transfer excitation internally and externally;
- A database together with ODBC the interface;
- Cards - objects of data presentation in the form of dynamic pages;
- Scripts – to support external reactions;
- Language environments Java and JavaScript;
- XML-grammars (dynamic and static);
- Integrator - a subsystem, capable to execute dynamic codes.

With the exception of Neural Core practically all components of this environment are freely available in the Internet and can be used for building applications on in different platforms - Microsoft Windows, UNIX or Mac OS. Some of them, such for example as *Microsoft Agent* or *Microsoft Speech Application SDK* for voice recognition and synthesis of speech, depend on operational system, and their application is possible only under control of Microsoft Windows. The technology *of Microsoft Agent* supported in Windows, allows using one more coordinate in structure of pages. *The agents,* existing as though outside of a surface of the document (Fig. 11) can represent the author and carry out a role of the virtual assistant for visitors the Web-site in the fashionable way.

Fig. 11 Three-dimensional design with Microsoft Agent
(http: // www.microsoft.com/msagent)

Unlike personal communication usually based on the *"point-point"* connection, the information amplifiers are capable to connect simultaneously with many sources of data, to find there the certain information and to give to its user in the most convenient form. Such solution is symmetric and similar to virtual representatives in the dispatching centers (Fig. 2) and both can be created with the same neural core components. Personal virtual agent can carry out a role of the individual secretary, being dynamically trained and being adjusted under the knowledge of the specific user and with a context date in his/her personal computer.

In applied programming there are different ways of developing applications - using of specialized environments (IDE, Studios), functional libraries (subroutines, classes) and others. *Excel* is a good example of the integrated environment and one of the most successful solutions in history of programming. In *Excel,* the specialized system of tabulated calculations is integrated with packages of subroutines of a different level which makes it ideal application environment, capable to solve extraordinary wide class of problems. There are several consecutive levels in *Excel* going from very simple, almost for everybody level, up to professional programmer's level:

- Creating sheets;
- Adding and updating data;
- Adding and updating formulas;
- Adding and updating of macro commands in Visual Basic;
- Integration with other systems through ODBC, XML, OLE, etc;
- Integration with other systems through DLL, C, C ++, etc.

If to use the similar organization of levels of programming in neural modeling, their hierarchy can look as follows:

- Creating neural layers;
- Adding and updating neurons and their links;
- Programming reactions on the excitation, incoming from outside;
- Addition and updating of macro-commands Visual Basic or JavaScript;
- Integration with other systems through ODBC, XML, OLE, etc;
- Integration with other systems through DLL, C, C ++, etc.

INTERACTING SYSTEMS

Let's consider the behavior of two applications - the calculator and the search system (Fig. 12). Both of them have few control buttons and input field. Both of them have the associated programs, which will be activated when input data are provided and button is clicked.

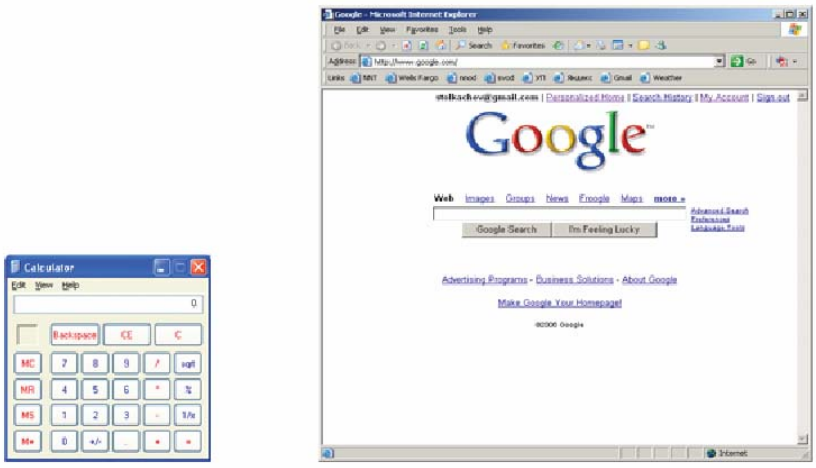


Fig. 12 Algorithmic calculator and not algorithmic search system

Our practical experience gives us the basis to assume, that for identical input sequences acting on various calculators, irrespective of time and a place, the result should be the same. In case of search system - we expect that the same chain of symbols on an input will bring most likely to us various results in various search systems and at various times.

Both the calculator, and search system are programs and as any programs, they consists of precise set of instructions which as is known, should lead always to identical result with identical initial data. It is obvious, that in case of search system the new result turns out each time as a process of changes of data which during execution of long sequence of commands have acted on an input of one of program modules. If we desire, we may using, for example, debugging and tracing tools, and having tracked all sequences of commands and events, precisely to define - where there was this change.

However the more advantage to us can bring the answer to a question: whether there are theoretical bases for distinctions in these two systems and if yes, whether it is possible to construct more effective technology and the environment of programming for the solution of problems of data processing in the Internet, considering these distinctions.

Let's assume that environment **U** includes all accessible to perception and analysis: objects, processes and events in the world surrounding us; and the system **S** is the part of environment **U** concluded inside of some border **B** (Fig. 13). The person, the computer or the organization -- are examples of systems. Any combination of systems in turn also can be considered as system.

Inside the environment **U** and inside the system **S**, there can happened events **E** which are defined by sets of parameters **P**. For each event **e** from **E** we shall assign the real number **P** in conformity. Events are very simplified representation of the changes occurring in the real world. We shall

assume that any change of a condition of environment or system happened as result of events. The perception of these changes in turn, too the result of events.

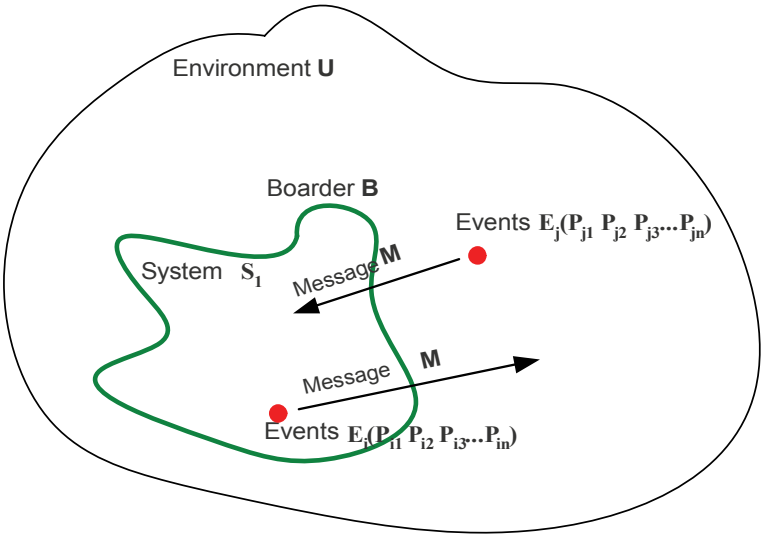


Fig. 13 Interaction between the system and environment

So, for example, the visual perception is excitation of neurons of an eye bottom or photosensitive elements of the receiver, is the result of reflection of light from the shined subject; the sound waves transferring speech which we can hear, came out as contraction of the certain group of muscles, as a result of this or that reaction of the person to irritation; electronic messages are initiated by the program started as a result of operation of the mechanism of interruptions, etc. For programmers, familiar with the Assembler and hardware architecture, systems of processing of interruptions can serve as a good example of event-driven programming, when data available for processing only after receiving the corresponding interruption (*IRQ*) which are signaling about data's readiness.

Any system in the real world is penetrated by immense quantity of various waves and corpuscular streams of radiations which constantly get through its borders and carrying in it huge quantity of the information. The man is capable to use only very small part of messages which can be filtered from these streams. If we could, having extracted from all streams of signals passing through our body, physiologically to feel as usual sound and vision all television and radio channels - our brain, most likely, was overflow and has given up during very short time. Considering the limited resources of brain, our ability to catch only the narrow range of electromagnetic waves is the vital restriction for the human being; however the same restriction does not allow us without special devices effectively to filter the helpful information from surrounding us dataflow.

Let define the interaction between the system and the environment in case, there was a change of condition in the system or in environment, as the message **M**.

If changes of a condition of system **S**, as a result of reception of message **M**, can be measured, we shall name such interaction as *informational*. We shall comprehend information **I** as in the certain way normalized measure of distinction **Δ** between states of the system **S** before and **S′** after the message received:

$$I \sim k * \Delta = S - S'$$

The opportunity to measure the information is the basic property, which allows analyzing interactions of systems. In some cases such measurement can be made according to the definition of the information, formulated by Claude Shannon in his work "*The Mathematical Theory of Communication*" [22.] Shannon suggested the model of communication system which consists of five components: *Information Source, Transmitter, Channel (with Noise), Receiver and Destination* (Fig. 14).

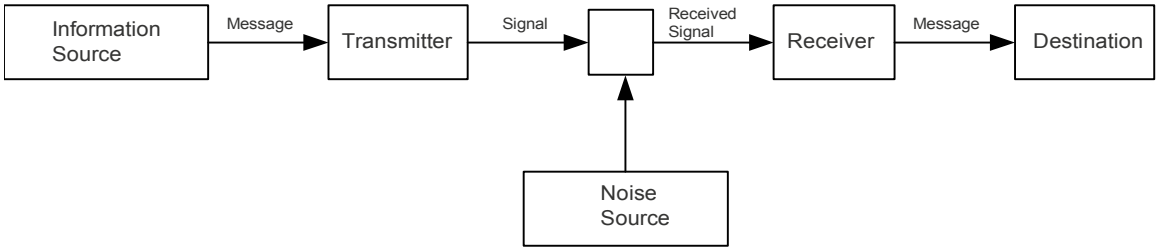


Fig. 14 Communication system according to Shannon

Shannon's work preceded by Hartley's "Transmission of Information"

That this scheme to apply to modern information systems, we need to make to it some additions. We shall assume that *the Source of the information* is a set of the pages placed on any site; *the Transmitter* is a Web-server, and *the Receiver* - a personal computer (Fig. 15). In this scheme the same basic elements of communication system Shannon, only their quantity are used all increases and, that is the most essential, the volume of the transferred information considerably increases.

Let's consider as an example inquiry in *system Google* on search of documents in which there is a combination of words - "*information and energy*". As a result we shall receive the list from references on more than 27 million pages! If to assume, that the required information which represents the answer to individual inquiry, can contain in any of these pages and consider, that the volume of average page in the Internet makes the order of 20Kb total length **L** of the message of **M** which all servers are ready will transfer in the channell for the subsequent processing in a personal computer:

$$\mathbf{L(M) = 2 \cdot 10^4 \cdot 2.7 \cdot 10^{10} \approx 5 \cdot 10^{14}\ Bytes}$$

At speed of the receiver 100 MB/sec it to be necessary it is more than year that one personal computer has managed to receive all pages. If, however, to increase speed of transfer by the order and to use at reception of this stream not one, but 100 processors, time, necessary to receive this message, can be reduced up to quite comprehensible, however abundantly clear, that the person thus is unable read through 27 million pages, with what speed they would not act on its desk.

Shannon considered channels with noise in which an element of data is the symbol that it is possible to compare to streams of the individual molecules flowing from one vessel in another under action of some force. Such model allows defining absolutely precisely physical characteristics of each separate molecule, but nothing speaks about a condition of all vessels as a whole. To speak about temperature, it is necessary to pass from molecules to volumes of gas.

As well the symbolical theory of the information - allows us to estimate precisely transferred data flows at an elementary level, but does not give a qualitative picture as a whole about the messages consisting of set of pages. Optimum conformity between physiological restrictions of a brain and characteristics of target interfaces to the information devices intended for an individual filtration

of data, can be based on speed of reading, which at people колеблется from two hundred up to five hundred words in a minute that corresponds approximately to one page of the text of the standard document or approximately three thousand symbols in a minute. It is possible to assume, that информативность documents should be based on other criteria, and first of all, it should consider specific features of the addressee. The quantity of the information containing in the document as a whole, and quantity of the information containing in symbols of this document, can not coincide and moreover, will necessarily differ for two various addressees.

Let's try to imagine a certain information measuring instrument which can give us the approached qualitative characteristics of the conditions similar to temperature of physical system. Such hypothetical device could answer a question, whether there is a sense to the person to read the next page from the list given by search service and, in more general form which of all set of pages can be read through. The answer to such question is possible, if we shall manage to enter some measure which will allow comparing individual human and machine representations about the information containing in messages.

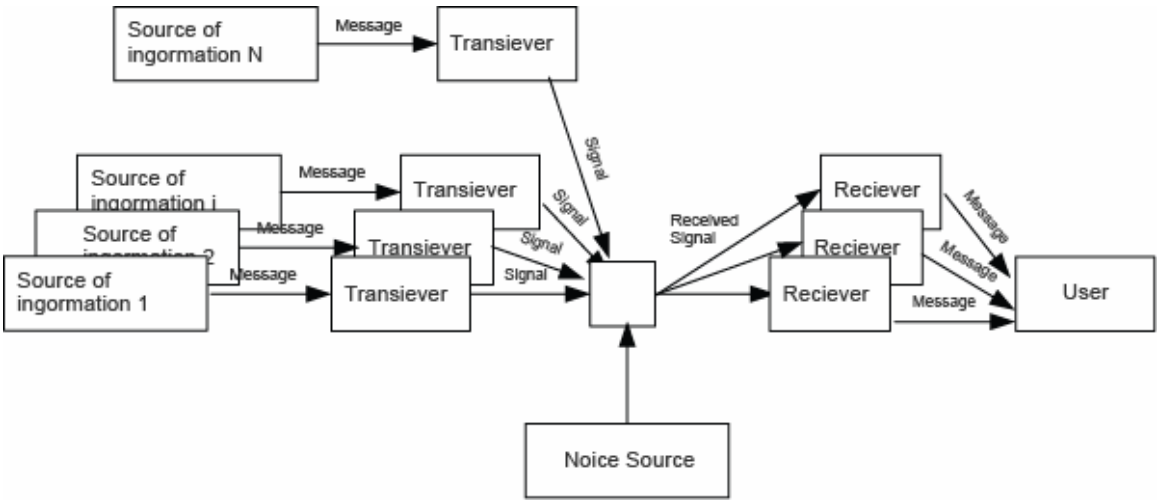


Fig. 15 Interaction in the Internet

The coordination of various ways of representations in programming is rather delicate problem also for the reason that the circle of users to which programmers should adjoin directly at creation of systems, is unusually wide. Terminology and definitions can essentially differ even in the event that it is a question of very fundamental concepts. For example, if in the American army terminological dictionary it is told: «*The information is the facts, data or the instructions which have been written down in any form and on any carriers*», and, in turn, consider the majority of mathematicians, that «*the Information is a measure of the variation of uncertainty about the object*» for the programmer it is senseless to argue and find out who is right - the army charter or the mathematical theory of communications. More important, following a known joke, having agreed both with the first and with the second definitions to develop the point of view which will lead to the most effective program decision. For this reason, in the further we shall use some compromise representations about concepts and criteria which will coincide not always with traditional for various scientific disciplines.

Among the works underlying theoretical representations about the information, it is necessary to allocate works of academician A.A. Kolmogorov especially. In clause «Three approaches to definition of concept “Quantity of the information”» Kolmogorov uses the theory of algorithms

with reference to definition «*quantities of the information, in something(**x**) about something **(y)** »* [34]. Kolmogorov suggests to estimate complexity, and so and информативность, objects in the minimal length of the program necessary for reception **y** from **x**. Being based on this approach, it is possible to understand the reasons of the problems which have arisen today before developers of search systems, and then to find ways of their decision.

Really, modern search systems have reached impressing results in classification of Web-pages. However the classification of information space is more deeply and more precisely spent, the more difficultly (and so and more long) the inquiry as a result of which, the user can obtain its interesting data should look.

So, if the search system will manage to divide pages in the Internet, into subsets which correspond to the certain criteria of classification (**A, B, C, D, E**...) to find its interesting pages, the user should know the list of these subsets and use it in the obvious or implicit form at the formulation of search inquiry. As a rule, pages interesting us are on crossing of several subsets and the logic of inquiry should reflect it (Fig. 16).

In the obvious form the inquiry about search can look as follows:

SELECT * FROM Pages WHERE A = x AND B = y OR C = z OR …

And in an implicit kind:

I am looking for a car, color = red, the manufacturer = Ford...

As a result, the search system should return the list of pages in which interesting data can contain. In an example resulted on Fig. 16, it there can be pages: 1, 3, 4. As the list of all criteria which the search system can use practically is never known, the user should apply the intuitive forms of inquiry which was resulting in which millions pages can contain.

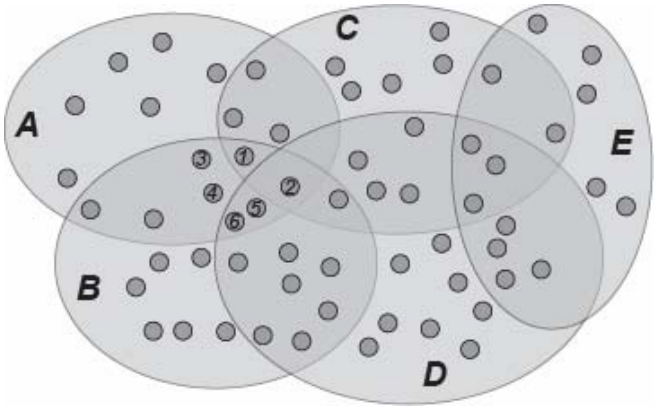


Fig. 16 Classification of pages in the Internet

It is obvious; that the quantity of pages in the Internet will continue to increase continuously as well as search systems will develop methods of classification. The decision of a problem of the formulation of search inquiry can be received by means of «*interactive search*».

In the example resulted above the server should ask such questions, as - *the color, what price, model,* and т. Item Kolmogorov's length serves thus to one of the main criteria of efficiency of

dialogue, and the problem of the designer consists in construction of the system capable for a minimum quantity of questions to give to the user the comprehensible answer.

INTERPRETATIONS AND MEASUREMENTS

We shall consider measurements in information systems, in many respects leaning on physical and, first of all, relativistic representations about such concepts, as a measure and the observer. The intuition prompts us, that individuality of perception is a natural property of the person which can be inherent as well in other systems. In computers ability to various interpretations of the same data has been initially incorporated in the structure which has received the name - von Neumann's architecture.

In architecture Neumann's background (Fig. 17) *the arithmetic-logic device*, *devices of management* and *input-output* exchange data through *memory*, writing down and reading out from it binary sequences. Interpretation of these sequences can essentially differ depending on the device: *the arithmetic-logic device* works with numerical or symbolical data presentations, *the device of management* - with commands, *devices of input-output* are interpreted by data depending on the form of their representation on external carriers. The same byte can be interpreted as a code of operation, binary number, the letter of the alphabet or brightness of a light point. It is possible to assume, that dependence of interpretation on the observer or, more generally, from system, it is kept at transition from a micro level - at work with bytes and symbols, to a macro level - at work with documents and other images of complex objects.

As already it was marked above, we shall be we consider only such messages which lead to change of a condition of system. We shall enter a number of additional restrictions and we shall specify, that we understand as change of a condition of system. Let with each event **E** inside of **system S**, the set of elements **N** is connected, each of which can accept some numerical value in a range from**-1** up to **+1**. In the further we shall interpret this set in the various images, but one of interpretations - probability, can be used for introduction of an information measure.

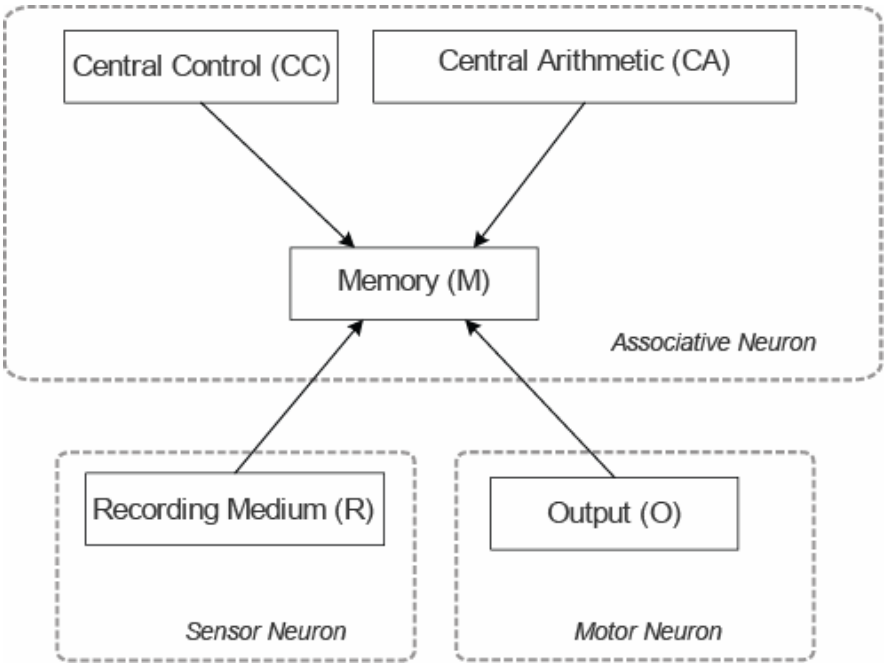


Fig. 17 Von Neumann's Architecture

Let's assume that there are some observers **O**, each of which can be both outside of, and inside of **system S** and to have the individual representations about events **E** (Fig. 18). We shall admit, that one of such representations of the observer about events is the likelihood characteristic of parameters of these events which can be set in the form of some subset **N**. We suppose, that two various observers can have various likelihood representations about same event **E** and that one and too the message can lead to various changes of these probabilities. We shall name such property - information релятивизмом by analogy with physical [8].

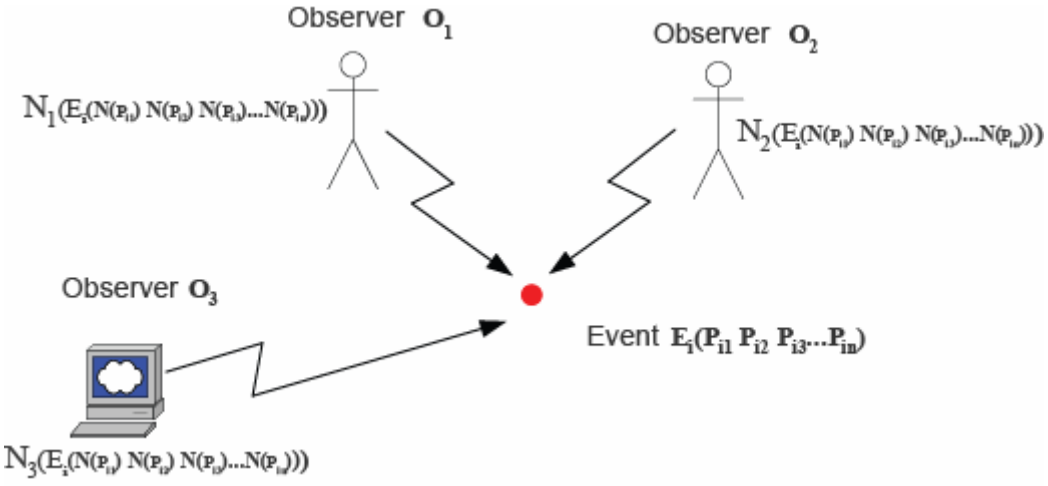


Fig. 18 Different probability of the same events depends on observation point

Let each of observers will receive sequence of messages **of M** which will change their likelihood representations about some event. Change of these probabilities in time can to look, for example, as it is shown on Fig. 19. We shall admit that there is some reference representation about probability of the given event (Fig. 20). Then we can enter concept of affinity of perception for observers which in a general view can be presented to each moment of time as distance between reference function **E (t)** and function of observer $\mathbf{F_i\ (t)}$*:*

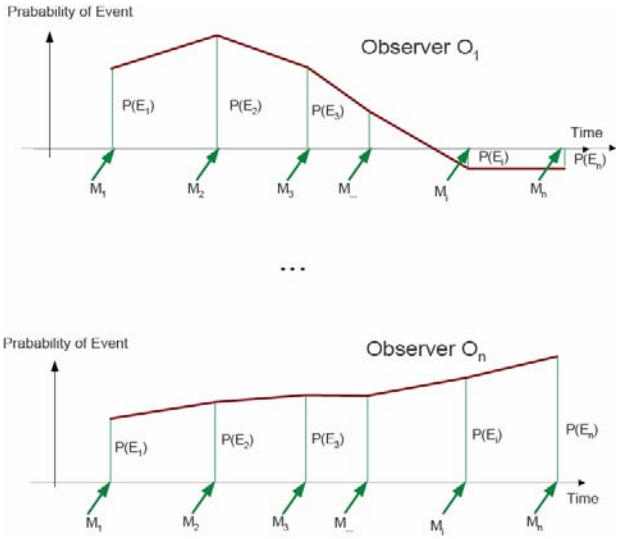


Fig. 19 Change in time of representations about probability of event at independent observers

$$D_i = E(t) - F_i(t).$$

The opportunity of analytical comparison of various user representations plays extraordinary important role in problems of electronic commerce in the Internet. As an example we shall consider technology of purchase of the car in virtual and real dealership. In the USA in 2004 it was totaled more than 20 thousand agencies on sale of cars in which works over 200 thousand sales representatives. The average wages of the sales representative make approximately $40 000 in a year. As small deviation, it is possible to notice, that this of three figures underlie economic motivation of all tendencies in electronic trade. Virtual representatives can lead to essential economy if they can execute a part of the functions traditionally fixed to the person. This tendency has led to that today the majority of agencies have a web-sites (Fig. 21) in which buyers can look through the databases containing lists of motor vehicles with full characteristics, including photos, the prices, opportunities of crediting, technical and operational specifications, etc.

In a database of concrete agency can contain from several hundreds up to several thousand motor vehicles. Using traditional methods of viewing and spending on the average one minute on inquiry, the person in a condition to estimate no more than 20-50 motor vehicles for one visiting of a site. Advantage from visit in real dealership in comparison with an independent choice on the Internet consists first of all in reduction of a trajectory of search and simplification of decision-making for the buyer. The real representative in agency aspires to understand as soon as possible criteria of the buyer and then in dialogue, varying parameters of these criteria, helps to pick up a suitable motor vehicle.

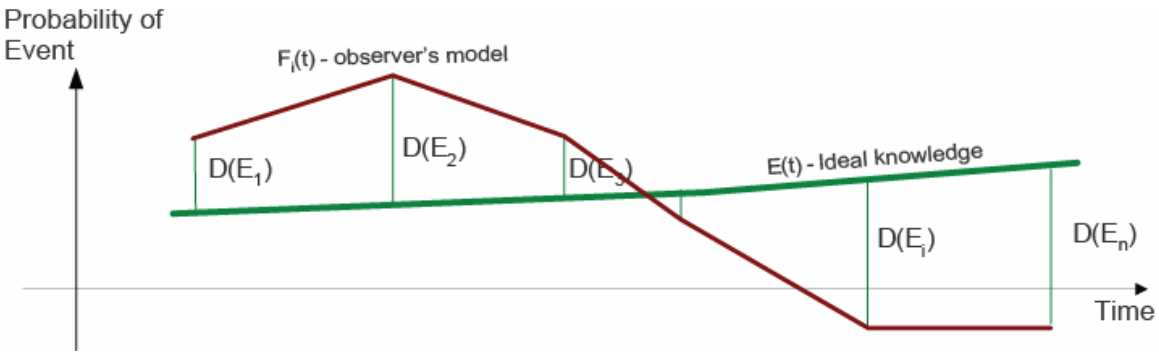


Fig. 20 Affinity of perception of the individual observer to reference knowledge

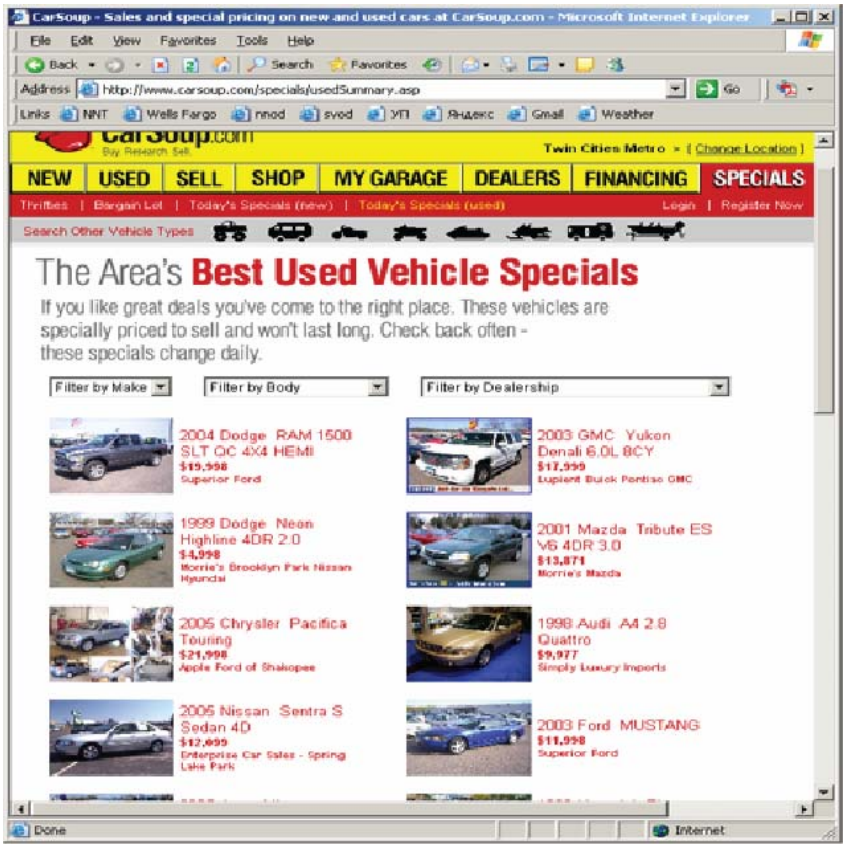


Fig. 21 An example of page with a database of one of agencies on sale of cars on the Internet

The model of this process is based that each car possesses some integrated characteristics, and each buyer has the individual representation about a desirable product. Process of a choice can be in that case shown to comparison of dynamic function of consumer representation with set of the characteristic functions defining the given kind of production.

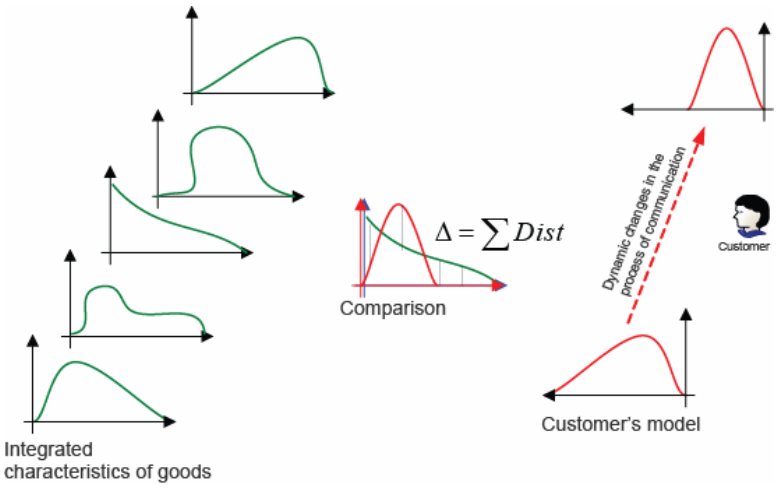


Fig. 22 Comparison of consumer's need with characteristics of goods

Strategy of the dealer is reduced to the *"best"* recognition of criteria of the buyer and then to addition to them by a component of the interest. Actually during informal dialogue with the buyer

the dealer catches and strengthens its representations and then compares with them to characteristics of motor vehicles which it has in the imagination (Fig. 22). Quality of the dealer, it is possible to estimate as its skill to understand and adequately to react to changes in wishes of the buyer; in its practical knowledge of specifications of cars, both in concrete dealership, and as a whole in the industry; and also in its ability to choose the optimum car on the basis of the dim representations of the buyer. In these processes indistinct definitions and criteria which do not allow receiving strict formal model of the decision as a whole contain many.

However in this enough simple, at first sight, to the scheme decisions which are key for creation of interfaces to so-called to "flexible *manufactures* » are applied. « *Flexible manufactures* » are intended for manufacturing production under requirements of the concrete consumer. The modern automated technologies allow reducing industrial expenses in the event that the formal specification of a product is known to a minimum. However reception of such individual specifications remains a serious and expensive problem. By means of interactive Web-interfaces this problem can be solved by creation of the virtual advisers assisting in drawing up industrial specification on manufacturing of production under the concrete consumer in conditions of mass industrial manufacture.

In modeling this process it is possible to allocate three basic stages - the task of characteristic functions, dynamic modeling of consumer representation and their comparative analysis between which there is a plenty as straight lines, and feedback [30]. The Feedback between result of the analysis and representation of the buyer is a key part for all models as a whole as the person constantly changes the purposes during search. It is necessary to note here, however, that in neural networks division of communications on direct and return has rather conditional character.

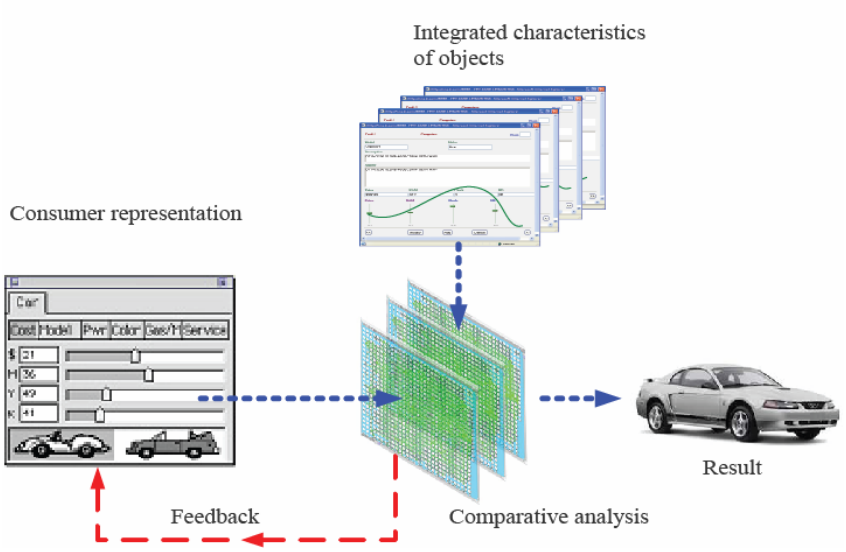


Fig. 23 Feedback approach to the search process

Registration of changes in representations of the buyer can be carried out or the buyer, for example by means of «sliders» (Fig. 23), allowing in the convenient form to set the approached values, or during dialogue with the virtual agent, with use of linguistic variables which can have such dim values as *"more", "more dearly, "more quickly"*, etc. The item some of criteria can have quite certain and exact values, for example the manufacturer or the name of a product - *"Ford"*, *"Toyota"*, "*Camry XLE*". Some representations can have numerical value, for example *«*

it is No more, than $10 000 » or *« No more than 100 000 km »*. We should unite all various kinds of these representations in some unified and constructive formation which will allow carrying out operations of addition, comparisons, changes and others.

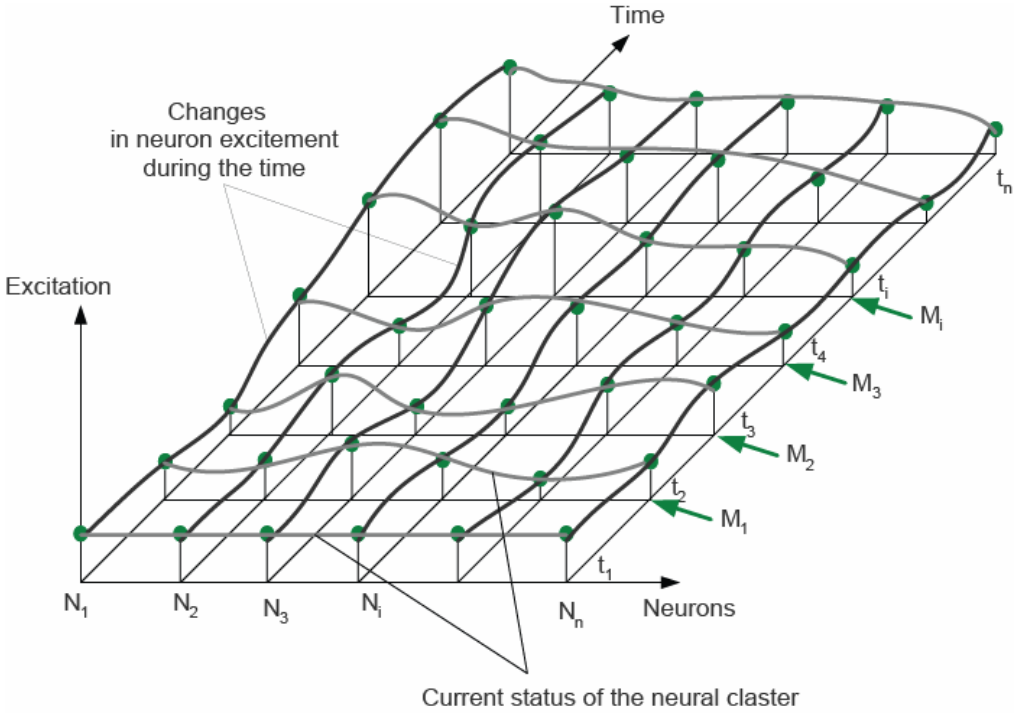


Fig. 24 Neurogramme - dynamic change of excitements of neurons

Such model allows to use mathematical methods of the analysis and possesses two important properties: on the one hand the mechanism of threshold excitation enables to carry out the certain actions, for example to start programs, on the other hand, current excitation of neurons can correspond to the dim approached values, characteristic for linguistic variables. It is obvious, that such model of neuron allows to store also and exact values, both logic, and digital binary and valid.

Let's assume, that all possible linguistic and numerical values can be displayed on set of neurons (Fig. 24) which excitation can change in time both as a result of external influences, and under action of internal mechanisms, such for example as *«forgiveness»*.

COMPUTING ASSOCIATIVE NEURONS

Let's address once again to model Neumann's background and we shall consider, that has occurred to initial idea of use of neurons as base elements for designing computers [19, 20]. Physiological systems contain huge quantity of the neurons connected with each other, each of which is quite finished computer. In modern computers operative memory contains set of the same elements, which quantity сравнимо with quantity of cells in biological systems, however all these elements are passive and unlike biological neurons are not functionally full devices. Originally a background Neumann has united *the device of management, the arithmetic-logic device and memory* in one block which it has named associative neuron or *the central processor* (Fig. 25). At first sight, its *processor* really reminds model of neuron, however associative neuron is not intended for storage of great volumes of data, and unlike a passive remembering element for it independent activity is essential. The computer on associative neurons so never also has not been realized - instead of set of active elements of which it should consist, following the initial scheme Neumann's background, a classical computer is constructed of the passive cells of memory connected with the unique *central processor*.

As a result modern computers have appeared as are far on the structure from physiological neurons, as well as the car from impellent system of the person. Thus both the car and a computer perfectly solve the problems - we can quickly move on transport highways and make effective transformations of data and calculations on known algorithms. It is possible to assume only as the computer facilities if associative neuron has been put in a basis of the COMPUTER would develop. As well as in many other things cases, the economy has appeared a determinative in a choice of a direction of development, and active neural components which a background Neumann assumed to use as base elements of the COMPUTER, have been replaced with more economic decision in which set of passive elements of memory work with one central processor that has in turn led to occurrence of consecutive methods of data processing and corresponding consecutive algorithmic languages of programming.

The first generation of neural models based on work the McCulloch and Pitts "*A logical calculus of the ideas immanent in nervous activity*" [18], has been constructed proceeding from the assumption, that nervous activity of biological systems submits to the simple law - « *all or anything* » and as consequence of it is « *neural events and parities between them it is possible to study means of logic of offers* » [18]. Artificial neuron constructed under this scheme contains two basic blocks - *the adder* and *the functional converter* (Fig. 26), and the network consisting of such neurons, is supposed constant in time. The further development of artificial neurons occurred in parallel to physiological researches on a way of addition of various functional converters, perfection of algorithms of change of weights of entrance signals, the return distributions, the plural interconnected layers and other properties.

Models of modern artificial neurons differ from McCulloch and Pitts first of all dynamic properties. The structure of neuron, including its entrance communications, the adder and the functional converter, can change in time. The neural network also is dynamic formation in which quantity of neurons and communications between them constantly change under influence of various factors. At modeling neurons in program environments we can use any variations of models - from the most simple binary, up to pulse self-transformed liquid neurons (*liquid neurons*) [15, 16], not having a stable condition.

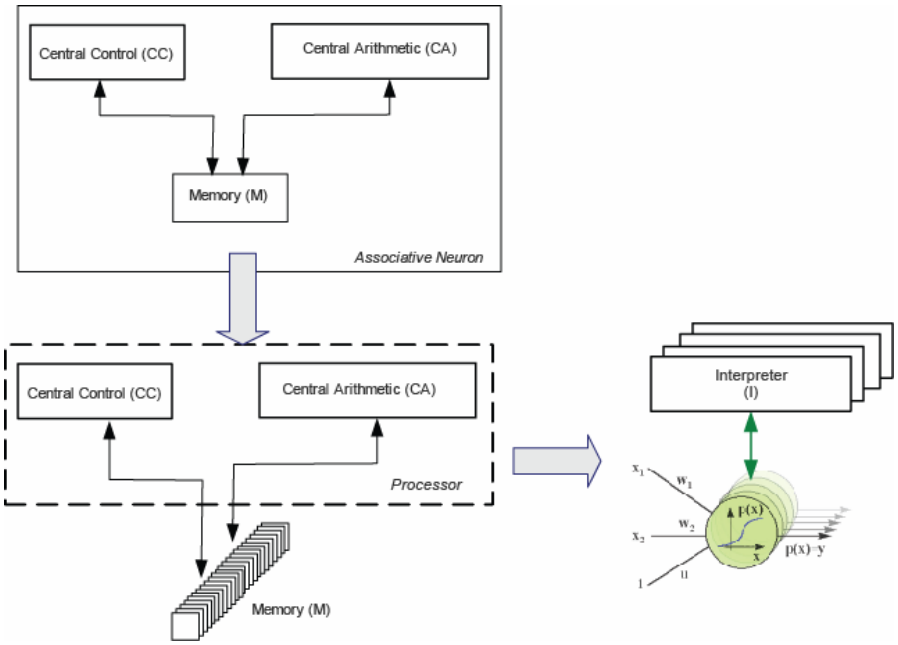


Fig. 25 Evolution of the neural architecture

As required we can create various types of artificial neurons, attaching executed scripts to concrete copies of neural objects. Program modeling of artificial neurons, unlike mathematical, first of all is engaged in researches of their constructive properties, and is far not last role thus economic feasibility and technological restrictions play. In abstract models connections of type « *everyone with everyone* » are quite supposed, that in many cases allows to simplify formalizations. In mathematical reasoning not essentially distinction between quantity of elements of set. However in practical realizations the size *n*, defining quantity of units in a network, can change all architecture of system radically. If thus to consider, that the quantity of communications grows as *(n-1) ×n,* becomes obvious, in what problems can result simple increase in quantity of units.

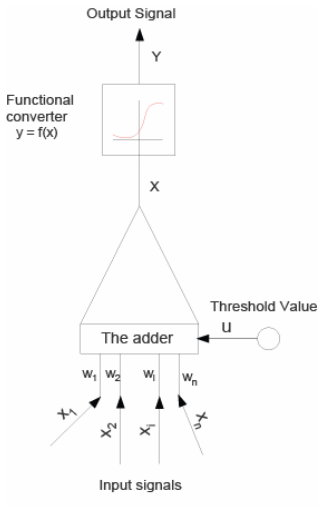


Fig. 26 The formal neuron

In practical programming the size of a file is one of the basic criteria influencing architecture of the decision. Depending on quantity of elements and properties of attitudes between them, such file can be created in operative memory, is placed in a consecutive or indexed-sequential file or organized in the form of structure of the certain database. It is obvious, that the choice of a way of a data structure is essentially important, first of all, from the point of view of a realizability of all model.

Generation or the tasks of elements of set directly are connected with way of a data structure. The majority of known neural models are isomorphic, based on neurons of one type. An opportunity to design polymorphic systems from various functional Neurons, certainly, will expand their functionalities, but on the other hand, will complicate process of their task and the subsequent development. Neural polymorphism will allow us to design the systems possessing essentially new properties in comparison with classical models. To similarly how DBMS have changed processes of processing and data storage, polymorphic models with the developed mechanisms of the organization and storage of neurons open opportunities for вариативной interpretations, creations of various generators and analyzers of a condition, access to them from a network the Internet and many other things of properties.

Addition of dynamic properties in neuron, cardinally changes all picture of modeling. The opportunity of internal changes, in a combination to external influences, is that combination of motive forces which allows neural models self modification. Really, if the system independently changes the condition (that somehow can lead to those or other reactions which have been not caused by external influences) the observer will register spontaneous, from its point of view, event, which occurrence is outside of its knowledge of determinism of behavior of the given system. Generally, enough two independent observers exchanging with system by messages that the behavior of system **S1** from the point of view of any of them, have got character unpredictable and independent (Fig. 27).

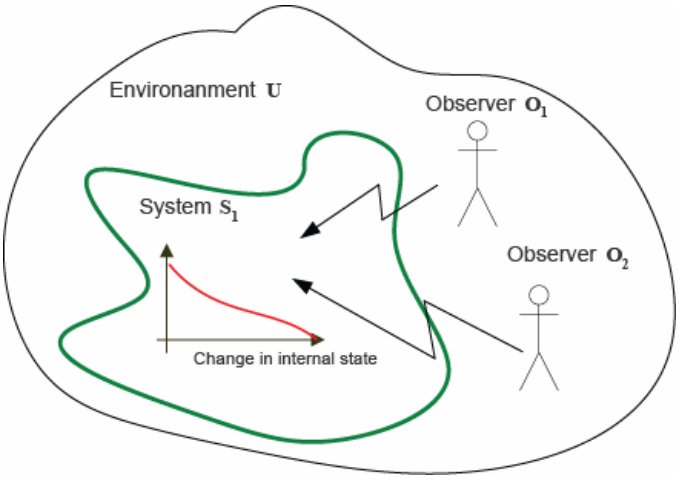


Fig. 27 Conditions for internal self-development

In traditional programming where the condition of variables is based on static methods, contents of a variable cannot change until the program will not carry out this change. Ability of neuron independently to change the condition is not essentially new in computer facilities - elements of analog computers behave in a similar way. However unlike rigid switching structures in analog computers which change is possible only at a stage of preliminary programming, neural models

are capable to add dynamically new elements, to establish new communications and to change their characteristics. Neuron as the element of memory possesses one more important property - it is capable to bear considerably a lot of the information, than 1 bit, stored in one binary cell of operative memory. The condition of neuron can be presented as a real number that essentially increases it information. If to admit, that biological neuron also is a remembering element with more than two conditions then estimations of volumes of data which the brain of the person is capable to store, should be reconsidered aside their substantial growth.

CONSTRUCTIVE NEURON

For us special interest the internal function chart of neurons, how many mechanisms of creation of neural ensembles represents not so much, their development and maintenance of internal processes of distribution excitement. Rather recent opening neurogenesis - process of a reproduction and restoration of nervous cells, allows looking on other at models of neural networks. That neuron during the certain moment could execute operation of division or has attached the output to inputs of other neurons, presence of some external factor *(interpreter)* $\mathbf{O_1}$ is necessary. It is possible to suggest, that the external observer can play a role of such factor, tracing a condition of system and cooperating with individual neurons, sending them the certain messages. For example, to connect an output of concrete neuron $\mathbf{N_i}$ to group of neurons {$\mathbf{N_k}$, $\mathbf{N_l}$, $\mathbf{N_m}$}, the observer can send neurons $\mathbf{N_k}$, $\mathbf{N_l}$ and $\mathbf{N_m}$ message $\mathbf{M_1}$ with the request to pass in a condition *selection* which will allow to allocate them from set of all neurons in system to which they belong. After reception of this message, neurons $\mathbf{N_k}$, $\mathbf{N_l}$ and $\mathbf{N_m}$ will pass in a condition *of a choice* that will allow the mechanism of connection of neuron **Ni**, to find neurons ready to connection and to construct communication of the output on inputs of the chosen neurons. This mechanism can join after reception of message $\mathbf{M_2}$ from observer $\mathbf{O_1}$. It is possible to assume, as in physiological neurons there is the similar mechanism operating development аксонов, and definition of a direction of their growth is set by a gradient of ferment or ionic *"smell"* which is allocated with target neurons (Fig. 28).

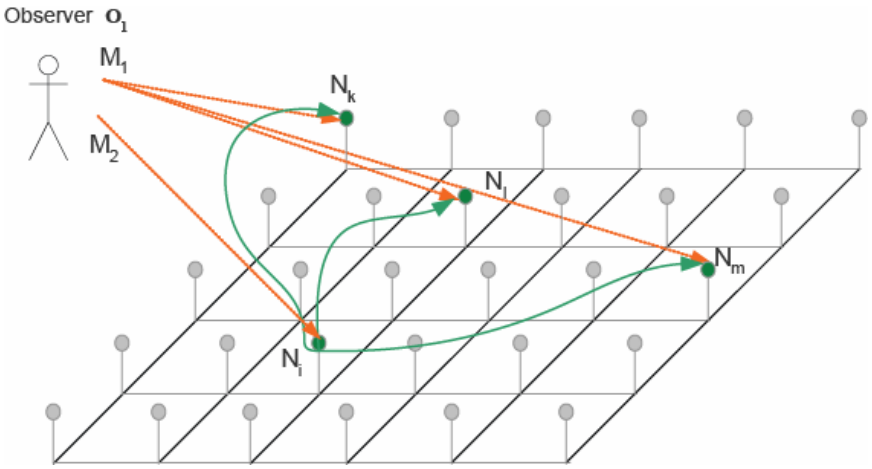


Fig. 28 Connection between neurons

If recognize that neurons can to be located in various areas of neural space, and also to have specific functional and structural properties, operation of a reproduction should create new copies in a vicinity of that area to which possesses the prototype, and also to contain the mechanism of changes which will allow to modify some of properties of its parent at creation of new neuron. The model of germination synapses in nervous cells is based on the assumption, that synapse grows in a direction defined by some field in which there is a gradient of the force setting this direction.

In the further we shall use demonstration examples which will allow the reader to experiment concrete program realizations of various neural models independently. The description of process of installation of the demonstration version *of Neural Experimental Environment* ***(NEE)*** is resulted in the *Appendix*.

On Fig. 29 the group of neurons from which green color allocates group of neurons (N_2, N_3, N_4, N_6, и N_{11}) with which establishes communication the neuron N_0.

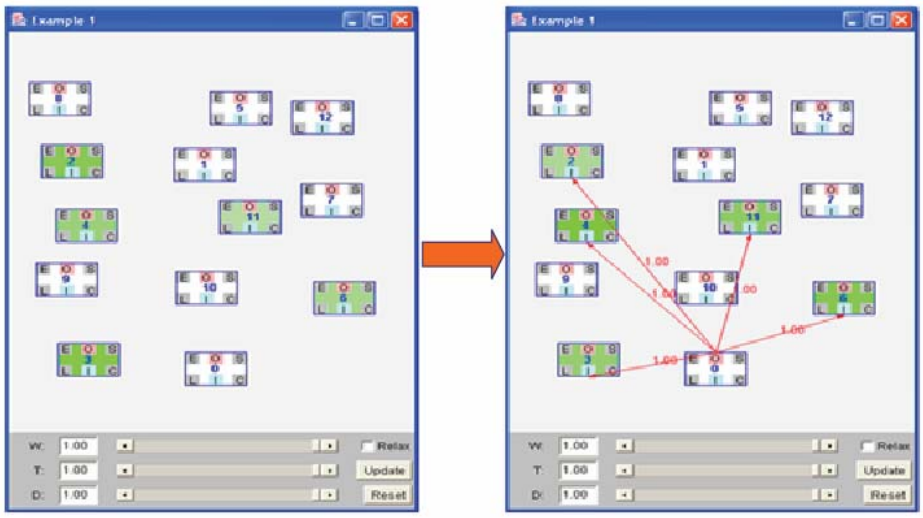


Fig. 29 Cloning and linking of neurons in experimental environment (http: // www.nnod.com/np)

Process of a reproduction is one of key in neural modeling and on efficiency of its decision depends how much successfully the systems constructed with its help can develop. In a basis of our neural model lays proto-neuron or the primary element N_0 possessing this fundamental property and capable to transfer it during cloning to all subsequent neurons. To start cloning, neuron should receive an external signal which can be initiated by the observer or any another, probably, internal object. Besides it, the observer can send neuron the message on change of the certain characteristics influencing reproduction of cloned neuron.

In our model each neuron possesses group of receptors - the interfaces, capable to accept excitation from an environment. Quantity of receptors and their functions can vary depending on complexity of artificial neuron. We shall consider, that our artificial Neuron has the minimal set of such receptors which are responsible for performance of the basic functions:

E - **E**xcitation,
S - **S**election,
C - **C**lone,
P - **P**ipe,
R - **R**eset,
T - **T**iming Clock,
I - **I**nput,
O - **O**utput.

Let's admit, that all messages which neurons with an environment can exchange, will contain the address of purpose and excitation - size which value lays in a range from **-1** up to **+1**. Such report unifies internal and external interfaces and allows internal neurons to send messages to other neurons in such kind which nothing will differ, from the signals acting from an environment. The unified representation of external and internal signals, and also the universal mechanism of perception, do possible internal development of neural system and at the same time allow reacting

to external influences. Differently, such model combines ability of internal self-development with external training under direction of the author.

The opportunity of cloning is one of the most important properties of biological organisms in the nature and also is one of the basic operations in object-oriented programming. The new object received as a result of cloning can or to copy in accuracy, or definitely to transform inwardness of initial object. For initiation of this operation in artificial neuron special receptor **C** which, in case of excitation, should lead to performance of operation of cloning is required to us. So for example, on Fig. 29 all neurons are received as a result of cloning initial *proto-neuron* $\mathbf{N_0}$.

For modeling dialogue interaction one more specific property - forgetting is required to us. During conversation or information search, the context is one of necessary conditions for reception of individual and quickly converging result. Context-dependent values can form complex associative groups for which we shall to use concept «forgetting» or function of the category *(discharge)*, allowing to change a level of excitation of corresponding neuron depending on time (Fig. 30, 31). This function plays the important role for many subsequent decisions; therefore we shall consider its properties in more detail. Real physical processes differ from ideal mathematical first of all presence of every possible additional dynamic component which influence their behavior. For example, inertial movement of any real material body is subject to forces of friction which finally will lead to its stop; the charged elements are unloaded eventually; the person can forget the facts or events of the past and т. The item this property of the real world is in the direct contradiction with main principles which underlie modern computers. The element base of computers and logic of programming are constructed, being based on the assumption, that value of a physical cell of memory or a variable in programming languages remains constant until any external actions will not change their condition. This static principle in many respects defines ways of realization and methodology of programming in which time is external property, in relation to objects of modeling.

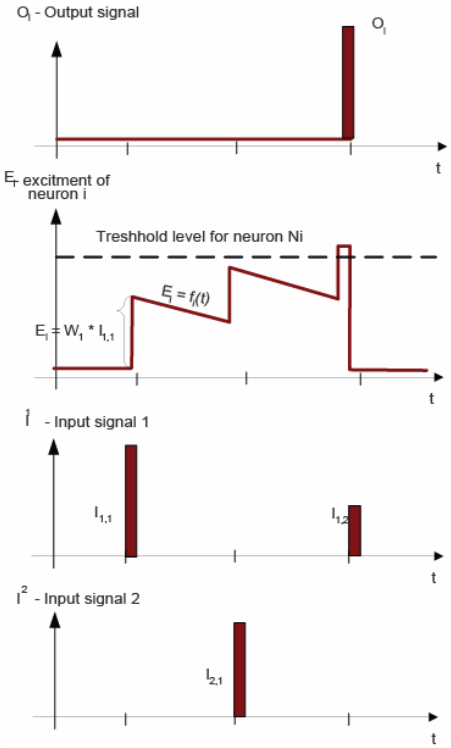


Fig. 30 The diagram of a state of neuron with two inputs

In our model we shall recognize that all conditions of neurons depend on time, and this dependence is internal property of the neuron. The static condition of neuron is considered as a special case when changes of neuron пренебрежимо are small and cannot be registered by the observer during some certain period of time. In this model of neuron all signals and all conditions have extent in time and their values can be нормированы concerning some base sizes.

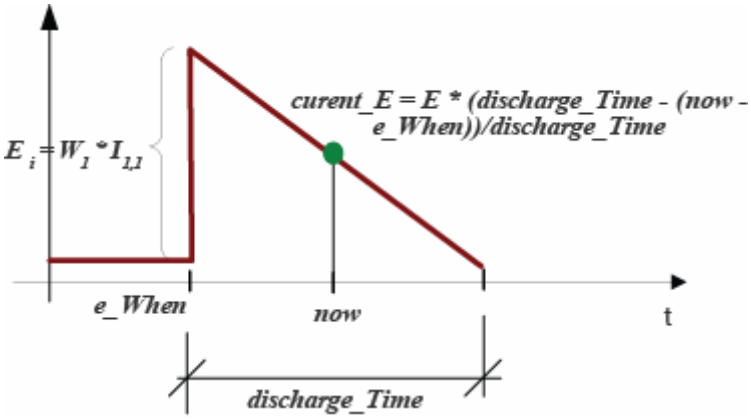


Fig. 31 The diagram of discharging of the neuron

So, the condition of our neuron can be defined as follows: in case of achievement of a threshold level, neuron *«fired»* the charge in all the neurons connected with it. In a normal condition neuron is unloaded and loses the charge depending on the factor of attenuation established for it - **discharge-Time**.

At each moment of time the current condition of the raised neuron is defined under the formula:

current_E = E*(discharge-Time – (now – e_When))/discharge_Time

Where:

E - a charge received by neuron from the outside during the moment of time **e_When**;

curent_E - a current charge of neuron in standard units from**-1** up to **+1**;

e_When - time in milliseconds during the moment when neuron has received excitation from the outside;

now - current time in milliseconds;

discharge_Time - time in milliseconds during which neuron will completely be unloaded.

For example, we can establish internal factor of a condition of choice **S** for 10 seconds, and after this time, neuron will return to the initial passive condition. Attenuation of neuron can carry out one more important function - generation of excitation after the certain period of time. We shall name this property - inverse excitation. Inverse excitation arises during that moment when neuron is completely unloaded, and its threshold of operation is established in **0**. This property allows using neuron as synchronizing element in the widest sense. For example, two in pairs connected neurons with inverse excitation form the multivibrator - the generator возбуждений with the set frequency. Single inverse neuron can serve as a logic element of a delay or act in a role of the generator of events during the certain moment of time.

On Fig. 32 the example of the multivibrator which allows to generate excitation on outputs of neurons «0» and «1» with frequency of 1 Hz and 0,1 Hz accordingly is shown. For these neurons threshold value is established in **-1**, for neuron of attenuation of "0" factor is equal 1000 millisecond, for neuron «1» accordingly 10 000.

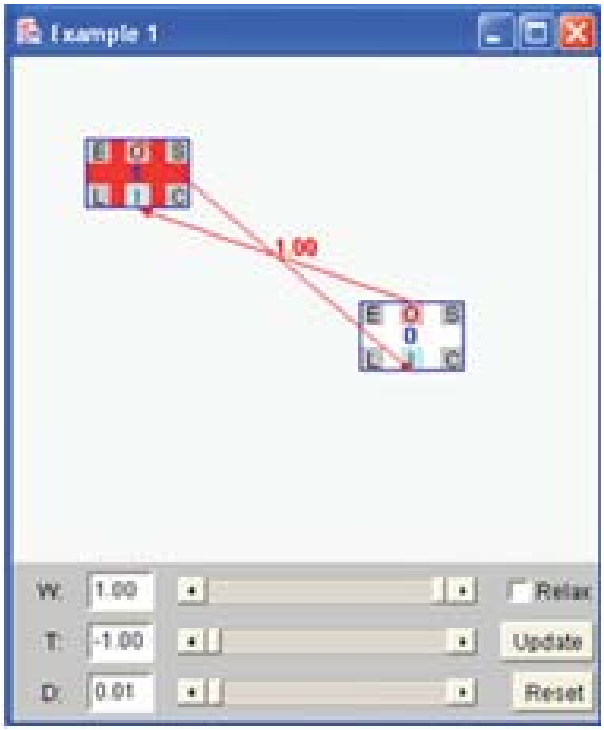


Fig. 32 The multivibrator on inverse neurons
(http://www.nnod.com/np)

Program models possess the specific features among which interpretability is one of the most essential. We can imagine the any abstract virtual computer which can interpret our model; however its concrete embodiment should satisfy to the certain criteria of productivity and profitability. By virtue of that our neural models are intended for work in the environment the Internet, the choice of a way of realization of artificial neuron is defined by mobility and simplicity of interpretation of its internal mechanisms, and as its interfaces with an external software/hardware environment. For today one of the most mobile languages for programming the Internet of appendices is *Java*, and leadership among protocols of an exchange, certainly, belongs *TCP/IP*.

On Fig. 33 the model of the separate isolated neuron having a set of variables which define its current condition and internal methods of interaction, including reception and data transmission under report *TCP/IP* is shown. Such neuron can be constructed rather simply in any environment of objective programming.

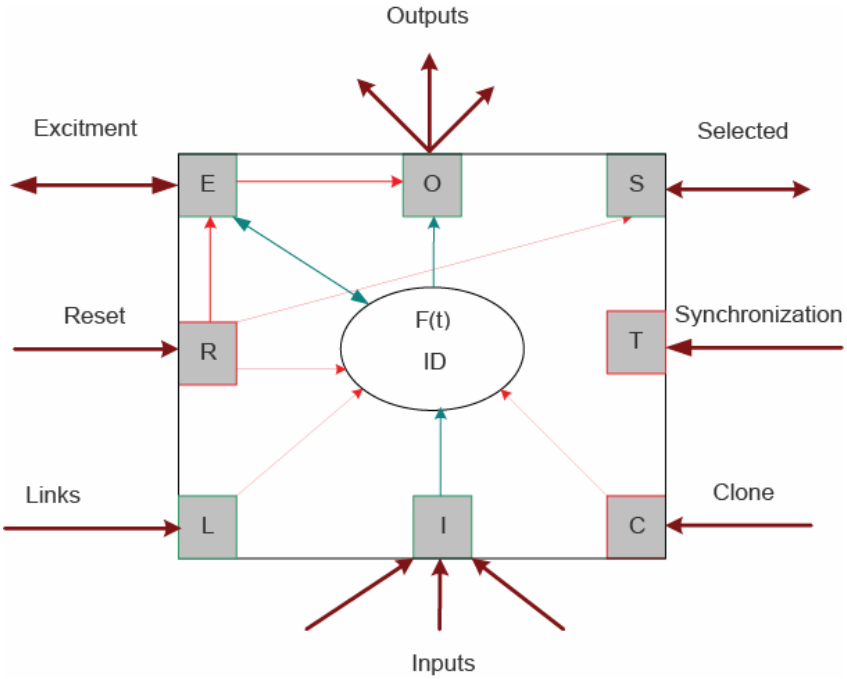


Fig. 33 Input/Output model of artificial neuron

Let's consider realization of this neuron in language *Java* (initial texts of programs are resulted on a site *http://www.nnod.com/np*). It can be presented as the object possessing internal methods which will allow it to exchange with an environment messages, using connections under report *TCP/IP* with application *Thread socket-Listener* in class *Node*. Receptors **E**, **S**, **C**, **P, R, T** and **I** can be realized as the ports opened for external connections at which the address can be calculated under the formula:

The address of port = Number of neuron ×10 + number of a receptor + displacement.

Such scheme allows constructing quite independent neuron, capable to carry out a maximum quantity of operations asynchronously with an environment. At all simplicity and enough close analogy to real neuron, this method possesses unique lack - its program realization will demand a significant amount of the resources allocated for ports, created in the moment of cloning of each concrete neuron.

It is possible to assume, that in the future this scheme will appear quite working, however in the subsequent constructions, we shall use more economic decision - all neurons in a concrete computer will cooperate with an environment through a network environment *(proxy)*, that will allow to reduce essentially necessary resources at insignificant complication of reports of an exchange with an environment.

Neural class *Node* and corresponding class *Nerve* in language *Java* has following structure:

```
class Node
{
        int id;
        double E;
        double Î;
        double D;
        double T;
        double S;
        Date e_When;
        Date o_When;
        Date s_When;
}
```

```
class Nerve
{
        int from;
        int to;
        String O_URL;
        double W;
}
```

Base methods for class *Node*:

discharge-Me *() { … }*
excite-Me*(double e-d) { … }*
fire-Me*(){… }*
link-Me*(){… }*
clone-Me*(){… }*
select-Me*(){… }*

The method **discharge-Me** *()* - is the basic way of change of inwardness of neuron. Real physiological neuron, or the neuron constructed by means of hardware elements, can have the internal mechanisms of synchronization which will allow making changes of a condition of parameters of neuron, time-dependent. In program emulators it is necessary to use external synchronizers which can provide a call of this method with the certain clock frequency that can be practically equivalent to hardware synchronization in the event that the interpreting computer will provide updating all neurons within the limits of one synchronizing step.

Thus restriction on quantity of simultaneously active neurons can be certain as follows:

Quantity of neurons < T/Δt

Where

T - an interval of synchronization,
Δt - time necessary on synchronization of one neuron.

Transfer of excitation to target communications of neuron depends on value of a variable **o-When**. If value of this variable is not certain or time of the category for output **O** is not a positive integer, we shall consider a condition of an output constant in time. In this case standard behavior will be such when in output **O**, there can be only one *token*. *The token* is withdrawn from output **O** in the event that for neuron is available, at least, one attached communication with other neuron or neuron has the internal reference to external port to which it will make a call, in case of achievement of a threshold level. Irrespective of quantity of communications, neuron will transfer in all external communications excitation which value is equal **+1**.

As processes of excitation of neuron and transfer of excitation to external communications are generally asynchronous, the situation when neuron can repeatedly be excited is possible before *the token* has been transferred in external communications. In this case there is an addition *of a*

token to output **O** which works as the store. Each subsequent cycle of transfer of excitation in an environment will reduce quantity *of tokens* on 1.

As processes of excitation of neuron and transfer of excitation to external communications are generally asynchronous, the situation when neuron can repeatedly be excited is possible before *the token* has been transferred in external communications. In this case there is an addition *of a token* to output **O** which works as the store. Each subsequent cycle of transfer of excitation in an environment will reduce quantity *of tokens* on 1.

Neuron can transfer excitation in an environment if at it value of internal variable **O-URL** is established. In a demonstration example the external module represents the small *JSP-program - neural.jsp* which should be placed in an accessible directory under control of Web-server *Apache Tomcat*. The call of this program from neuron will lead to its start and as result, this program can generate as much as complex process-reaction.

At the reference neuron will transfer in the caused program unique parameter - number. The same program can be caused by usual way:

http://www.nnod.com/np/neural.jsp?ID=12345

Below the initial text of this program is resulted:

```
<%
        String ID= request.getParameter("ID");
        if (ID == null) ID = "";
%>
OK ID = <%=ID%>
```

This example is intended for demonstration of work of neuron in « a motor mode » when result of its work can be excitation of the external program module (Fig. 8).

For convenience of designing of neural models, besides classical communications providing transfer возбуждений between neurons, we shall enter an additional class of structural communications. We shall name connections between neurons on which signals of excitation can be transferred *by active communications* $\mathbf{P}_{i,j}$ *(Pipes)*. *Structural links* $\mathbf{SL}_{i,\ j}$ (*Structural Links*) serves for allocation of groups of the neurons possessing certain properties. For example, group of the neurons representing word meanings of the concrete dictionary, or group of the neurons representing distinguishing diagnostic layer, etc. Structural communications allow setting easily, and accordingly to allocate from all set of neurons of the given model only what belong to the certain layer and by that to create the multilayered interconnected structures. In itself *structural links* do not render any influence on distribution возбуждений among neurons and carry out exclusively auxiliary role, facilitating understanding and simplifying process of designing. We shall name group of the neurons belonging given hierarchy - *cluster* (Fig. 34).

Presence of structural communications allows using traditional methods of work with the structured data. Well-known in relational databases of operation of sample can be to the full применимы to кластерам in neural models. Moreover, integration of neural models and relational databases allows simplifying essentially the decision of a plenty of the problems connected with the organization of neurons in various groups, depending on interests of the concrete user.

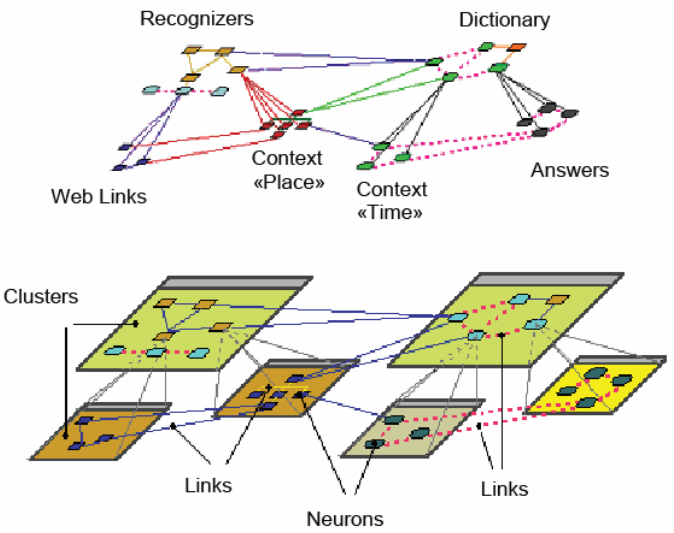


Fig. 34 Structural links and their representation in the form of clusters

Structural links can be set in the form of the relational table:

Name	***Type***	***Description***
ID	*AutoNumber*	*Unique identifier*
Cluster-Name	*Text*	*Name of the cluster (may be empty)*
From	*Number*	*The identifier of neuron, to which belongs data cluster (may be empty)*
Neuron-ID	*Number*	*The identifier of neuron, which belongs to given cluster*

The neurons belonging to certain cluster, it is possible to choose by means of standard *SQL-inquiries*. For example, if exists кластер *'Rus-Dictionary'*, the set of neurons belonging to them can be received as follows:

SELECT Neuron-ID FROM Clusters
WHERE Cluster-Name = 'Rus-Dictionary'

Structural communications allow applying visual graphic interfaces and the conceptual methods of designing similar to what are used in the theory and practice of databases at creation of neural models. In practical designing databases the success is frequently reached due to a successful combination of various methods of modeling: mathematical, conceptual, logic and physical ways of data presentation, their attitudes and functional interfaces. Conceptual representations in this number take the special place as unlike the mathematical models intended, first of all, for reception universal and the common decisions, in conceptual models distinction in the points of view and interpretations of different people is considered at work with the same data sets. Problems of the designer thus include revealing these distinctions and development of mechanisms of interpretation and data presentation, depending on their concrete consumer. Conceptual representations and information релятивизм can be carried to the same phenomenon - identical messages are perceived and interpreted by various systems on a miscellaneous.

NETWORK SHELL AND BASIC PROTOCOLS

Sensor controls of neurons provide perception of the signals acting from an environment. For their modeling we shall use two types возбуждений: direct excitation in the event that the graphic interface is used, and network excitation on the basis of report *TCP/IP*, in case of when the signal comes from any removed source. The graphic interface and interaction through the Internet-reports form system of access, well-known to experts on databases where inquiries in the graphic form (***Query By Form*** on the one hand are used), and with another - programmers can always use language *SQL* as a universal remedy of construction of any references to data. As well as in databases, visualization of neurons and communications between them can have set of various forms, depending on the problems solved by the concrete user.

Training and designing are the same important properties in neural modeling, as well as reaction of neuron to external irritations. Sensor controls and receptors thus simultaneously are carried out with two essentially various functions - accept signals which conduct to change of structure of neural model and react on external excitation which lead to response that it is quite possible to compare to process of programming and execution of programs. Programming changes structurally functional properties, and execution of programs consists in reaction to the data acting from the outside. In neural models programming and execution are actually incorporated in uniform process of an exchange возбуждениями. Thus concept *the program* is washed away, and we can speak only about changes in a condition of neural systems during their interaction with an external world. Such property of neural models quite corresponds to our household representation about human behavior when training and reactions during dialogue practically are not divided - whether is received we new knowledge or we answer questions which to us well-known, it occurs in a context of continuous inseparable interaction during all human life.

As we already marked above, changes occurring as a result of interaction in a condition of neural system can be easily enough measured and shown to quite final set of the values reflecting a current condition of neurons and their communications. Being based on such measurements, we can use methods of the quantitative analysis for definition of characteristics, as dynamics of development, and as whole complexities of neural systems. Comparing behavior of molecules in physical systems with behavior of neurons, we can notice, that neuron possesses practically same quantitative set of characteristics. The moment of quantity of movement of molecules and a level of excitation of neurons can form the basis for their comparison that allows using physical analogies with reference to the information analysis of neural systems.

In the graphic version of the user interface included in *structure NES*, interaction with neuron it is carried out by *"touch-click"* in area of the certain receptor that will lead to transfer of a signal to neuron for its subsequent processing. Such way is evident enough for studying properties of artificial neurons and can be used at construction of base neural structures.

As required we shall consistently add new opportunities in the report of an exchange with the neural environment. The first level of this report practically corresponds to intuitive physiological model - excitation of a receptor occurs in case of occurrence of a signal on its input that can be set by means of two parameters: the address of a receptor in neuron and a level of a signal. Neuron receives a signal on an input after it was transformed by a nerve (communication) according to factor of transfer. However if on an input of neuron the signal from an external source acts, the level of excitation should be obviously set by the transferring party. In this case the following format of the message is used:

The message: =
<GET> </number of neuron> = <receptor> |
<GET> </number of neuron> = <receptor> = <size of a signal>

For an exchange with the Network environment (NS) it is possible to take advantage of any telecommunication program which supports connections under report TCP (Fig. 35). The most simple and accessible way of transfer of a signal on a receptor of concrete neuron is the HTTP-inquiry.

Practically in all libraries of the modern languages of programming there are means for creation *HTTP* of inquiries, no less than in the majority of applied appendices - *MS Excel, Word* and т. The item is the built in means of hyperlinks which can be used as the initiator of such inquiry. The simplest way of transfer of a signal on a receptor of concrete neuron is activation of a hyperlink in one of such appendices, for example in the Web a browser.

Each network environment represents the independent program and we can have any quantity of simultaneously working spears, having appropriated each of them own port. As each port, has unique number, in the further we shall use this number for identification of network environments and, accordingly, neural structures located inside. The reference to an environment for which *the port 10000*, for example, *will be allocated* to look as follows:

http://localhost:10000/0=s

At reception of such inquiry will allocate from a body *of HTTP-inquiry* a command, will define type of a receptor and will execute corresponding operation with concrete neuron.

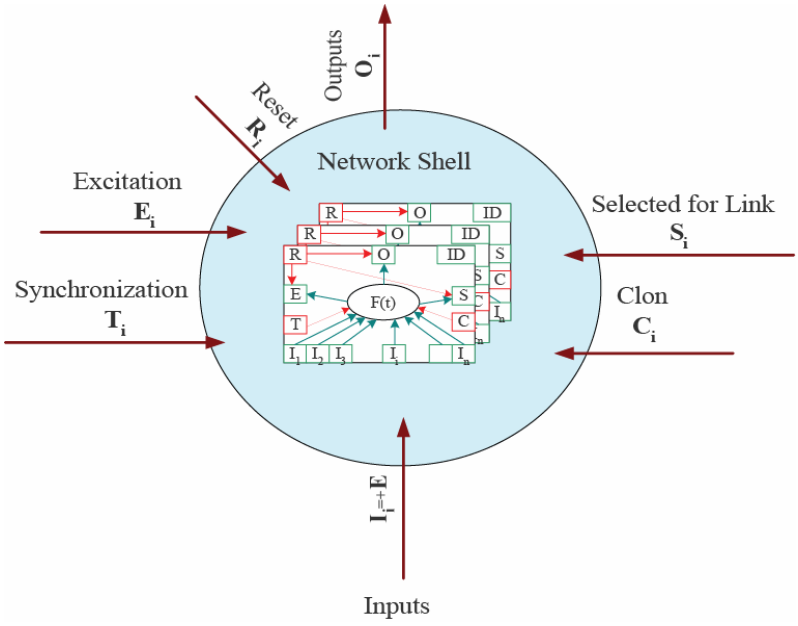


Fig. 35 Network shell of artificial neuron
(i - number of neuron)

That the reference to did not lead to a constant reload of contents, the additional frame is required to us (Fig. 37), will be carried out whence *a http-call* and there the received result will be returned.

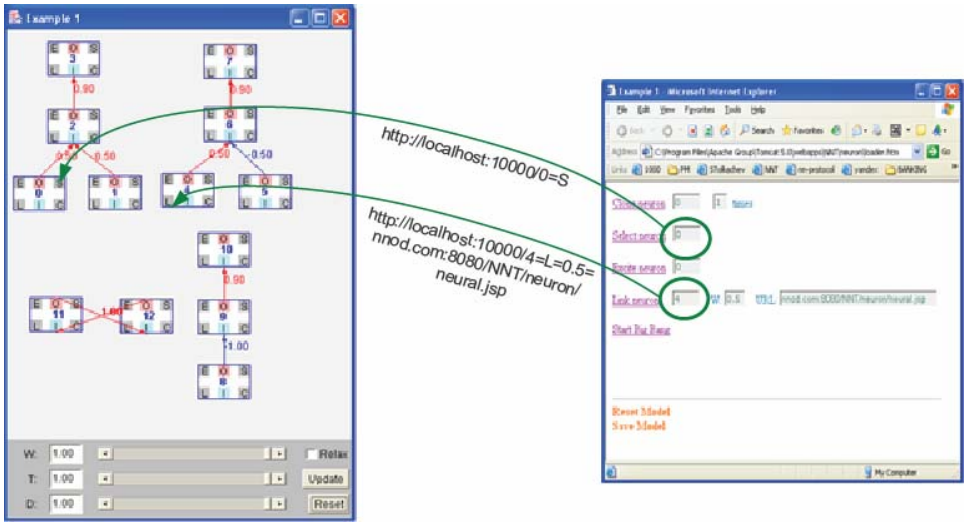


Fig. 36 Neurons can perceive signals from the Web.

On Fig. 36 the example of interaction of neural *environment* about a web-page which is a part of a demonstration example is shown. On this page some hyperlinks to neurons with use of dynamic properties *DHTML* and language *JavaScript* are created.

That the reference to did not lead to a constant reloading contents, the additional frame is required to us (Fig. 37), will be carried out whence *a http-call* and there the received result will be returned. We shall use this frame for execution of the programs dynamically loaded in given page from various external sources. Such opportunity appears at us because the page in this case is quite independent *interpreter*, and there are some interpreted languages (*JavaScript, VBScript*, etc.) which possess sufficient properties for maintenance distributed computing processes.

Such two-layer pages consist of the visible foreground which we shall name *a card (Card)* and the second latent layer - *a background (Background)*. The multilayered structure does possible effectively to use a web-page as the active program component, allowing to carry out in the latent layer all necessary actions on a call of external systems, processing of the received results and their transfer for display in a visible layer. Call from this frame in language *JavaScript* looks as follows:

```
Host = "localhost";
Port = "10000";
function call-Neural-Environment (Neuron-ID, op)
{
URL = Host + Port + "/" + Neuron-ID + "=" + op;
parent.fControl.location.href=URL;
}
```

As the external interface we can use server *Apache Tomcat* that allows through *ODBC/JDBC* to coct the interface our neural environment to databases or to other functional appendices. Instead of *JSP/Servlets-technology* with the same success the technology Microsoft can be used *ASP*.

In the further we shall use multilayered pages for management of the neural environment, an exchange with a database, the organizations of dialogue and other functions. Such organization allows constructing with the minimal expenses on the basis of a web-browser practically

unlimited, open and expanded set of functional components in which powerful graphic means are combined, opportunities of dynamic programming and the developed interfaces to a wide spectrum of systems and applied programs.

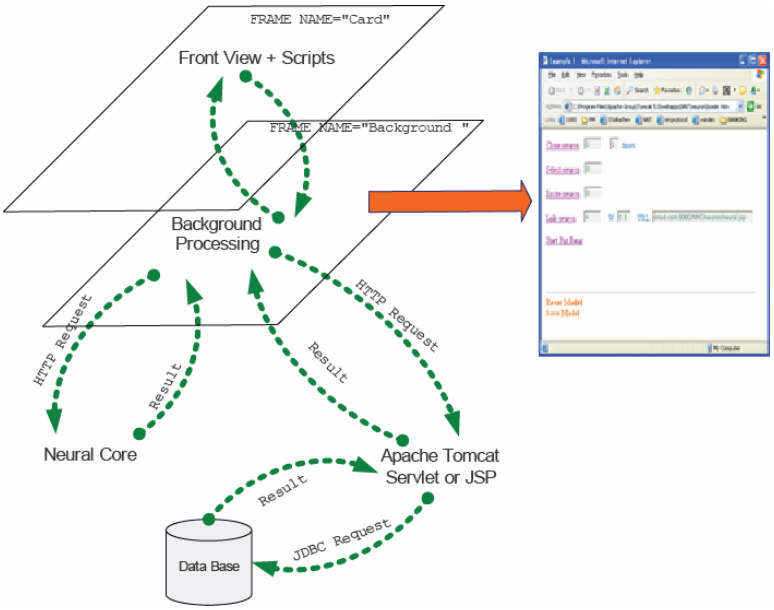


Fig. 37 Multilayered (sandwich) structure of DHTML-page

In loading of frames is carried out through intermediate page - *loader.htm* which contains a following code:

```
<HTML>
<HEAD>
<TITLE> Example 1 </TITLE>
</HEAD>
<FRAMESET COLS = “*”>
<FRAMESET ROWS = “ 100 %, 0 % “>
<FRAME NAME = “Card” TARGET = “Card” SRC = “neuron.htm”>
<FRAME NAME = “Background” TARGET = “Background” SRC = ““>
</FRAMESET>
</FRAMESET>
</HTML>
```

Programming of neural appendices in languages of type *Java*, allows using all modern means and technologies for integration of these models into various applied systems. Thus we can coct to them such structures of data as dictionaries, thesauruses, index files, etc. We can expand the report of an exchange with neurons, having entered, for example, additional designs in language *SQL* which will provide management of cloning, возбуждениями, communications and other operations, implemented in neural models:

```
SELECT NODE/LINK ... FROM ... WHERE ...
UPDATE NODE/LINK ... WHERE ...
DELETE NODE/LINK/CLUSTER ... WHERE ...
CLONE ...
```

LINK ...
EXCITE ...

By analogy with *ODBC/JDBC* commands of an exchange with the integrated neural environment within the limits of the report of a high level can be built in language as follows:

```
query = " SELECT NODES " +
" FROM * WHERE ID = 123 ";
ns = neural-statement.execute (query);
more = ns.next ();
while (more)
{
        ID = ns.getString (1);
        more = ns.next ();
}
```

Application of this report at programming *modules JSP* or сервлет under control of *Apache Tomcat* allows keeping the general style of an exchange both for databases, and for neural models.

Touch and motor properties of neurons in our case are shown to an exchange of messages through the Internet, and associative neuron, having kept all the basic properties of static artificial neurons, a number of the dynamic properties expanding its functionalities has.

At construction of program model of artificial neuron we have an opportunity to take advantage of means of traditional object-oriented programming languages, such as *Java* or *C ++* or, considering necessity of storage of models for a kind independent of the concrete interpreter to use objective databases, such for example as *IBM Cloudscape*, that are more suitable environment as in them, probably, to create active models, integrating data, objects and processes within the limits of one system. However as objective bases at the moment still are at an experimental stage of development, we shall use the conceptual positions incorporated in them on integration of methods and data, but to keep our models in traditional relational bases, for example - *MySQL*.

For preservation and restoration of neural models, and also for work with accompanying data, such as texts, dictionaries and reactions, in a demonstration example we shall use *ODBC* the interface that will allow us to create the decisions independent of a concrete database. Installation and configuration *ODBC* does not represent special difficulties in modern operational systems.

In table 2 the example of structure of a database which can be used for storage of neural model is resulted.

Table 2

Name	***Type***	***Description***
ID	*AutoNumber*	*Unique identifier*
Name	*Text*	*Name of neuron (it can be empty)*
E	*Number*	*Value of internal excitation (+1 <>-1)*
O	*Number*	*Value on an output of neuron (+1 <>-1 or it is empty)*
E-When	*Date*	*Time when neuron has received next Excitation*
O-When	*Date*	*Time, when neuron, having reached a threshold*

		Operations, the course has changed
D	*Number*	*Value of factor of the category*
T	*Number*	*Value of a threshold of operation*

NEURONS, WORDS AND SENSE

Since ancient Greeks [1], problems of language, understanding and interpretation draw attention of many generations of researchers. In programming, this direction certainly is one of essentially important as all theoretical and practical bases of this discipline are connected with concept of language. Classical definition « *Language* **L** *is a subset of set of words of set alphabet* **V** *which can be set by means of grammar* **G** », in this or that form is familiar to the majority of professional programmers. Proceeding from this definition, process of translation and performance of programs that does their *"understanding"* at a computer level quite certain and unequivocal is formalized. However application context-free грамматик, having great value for formal programming languages, in the problems connected with interaction in natural languages, as a rule, does not bring positive results. What theoretical reasons would not stand up for it, the successful practical decisions based on application formal грамматик in a class of problems, connected with dialogue in a natural language, practically do not exist. Under successful we mean the decisions accessible to the applied programmer, having an operational experience with standard languages of programming and resources of an average personal computer.

Attempts to construct effective systems for the decision of the problems connected with understanding are undertaken during all history of programming. It is enough to name such languages as *LISP*, *PROLOG*, *Smalltalk*, systems - *Elisa*, *MICIN* and others. The bibliography on this theme is rather extensive (see, for example, [31]). Our purpose consists in that, whenever possible making use this experience, to learn to create simple, but at the same time the useful systems, helping to solve problems with which the majority of programmers collide daily. For this purpose we shall consider a number of examples which will allow constructing as a result the finished system at which there will be all basic aspects of understanding and which will be capable to communicate with the user on some subset of a natural language depending on a concrete situation.

Let's consider the organization of the elementary communications between the neural environment and *words*. *The word* is a basic element of construction practically all linguistic systems. We shall take advantage for its specification by definition of the term *a word form* (*word form*) in *system WorldNet* [9]. Not going deep into research of formal properties of this concept, we shall consider that in our case *the word* is the sequence of symbols limited by a blank or other special symbol which can be allocated from the text by means of the certain program. Us the origin of the text will not interest - it can represent the program or a part of a literary work, the message in a chat or the sequence received on an output of system of recognition of speech. For us there will be quite enough that there is a program which will manage to allocate from this text of a word. Operation of allocation can be used for the most various purposes, for example for creation of dictionary **D** or for excitation of the neuron corresponding this word. For example, for allocation of words from the text and construction of the elementary dictionary it is possible to take advantage of the program *(JavaScript)* resulted below:

```
var dictionary = new Array();
function extract-Words(raw-text)
{
        words-array = raw-text.split(" ");
        words-array = words-array.sort();
        j = 1;
        dictionary[0] = words-array[0];
```

```
        for (i = 0; i < words-array.length-1; i++)
        {
                if (words-array[i] != words-array[i+1])
                {
                        dictionary[j] = words-array[i+1];
                        j++;
                }
        }
}
```

In the improved version of this program in which to the program the method of removal of dust and editing of special symbols (Fig. 38) is added is applied slightly.

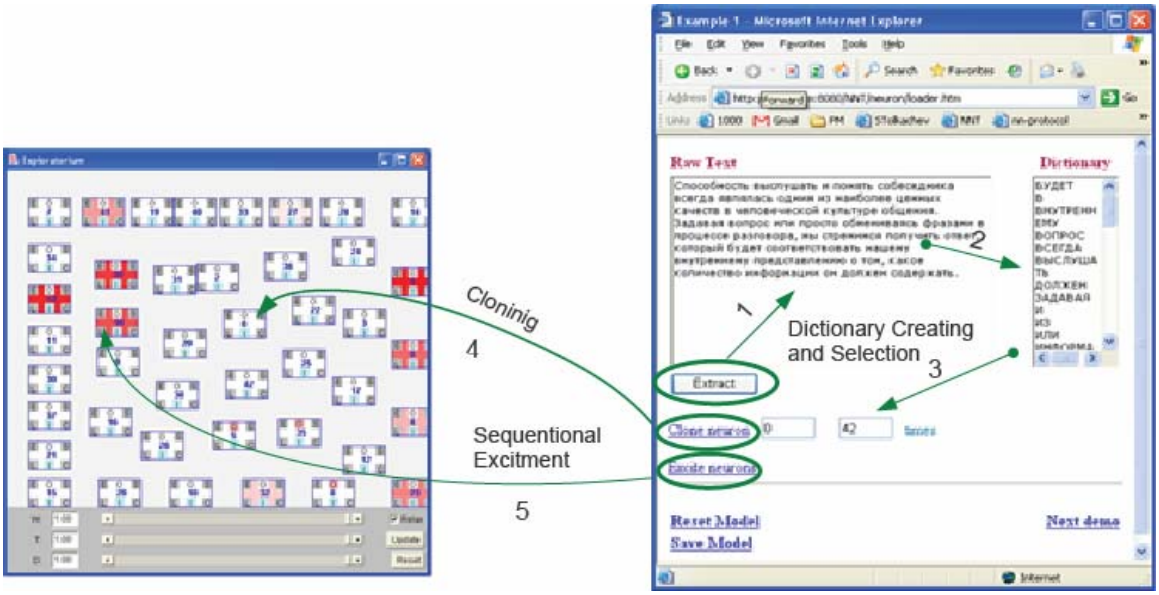


Fig. 38 Links between neurons and words
(Example 2 http://www.nnod.com/np)

Result of work of the program, after text processing, dictionary **D** consisting from *n* of unique words $\mathbf{W}_i$ is. On the basis of this dictionary we can construct neural layer $\mathbf{L}_1$ consisting of *n* neurons $\mathbf{N}_i$, each of which corresponds to word $\mathbf{W}_i$ in dictionary **D**.

For creation of the neural group consisting from *n* of elements, we shall take advantage of operation of cloning and we shall apply it, for example, to neuron $\mathbf{N}_0$. On Fig. 37 the example of sequence of actions for this decision is resulted. At the first stage there is a construction of the dictionary from words of the any text which in the further is used for definition of the index belong to this dictionary of a word. In the further excitation of neurons can occur in a mode of consecutive reading (*Excite from text*) or is selective - in case of allocation of any word in the text.

Within the limits of one neural model, the any quantity of neural layers and any quantity of network environments can be constructed. Construction of multilayered models is made by designing and allocation of neural layers and an establishment *of structural communications* between neurons for their subsequent visualization. In this example we shall assume, that one

neural layer in accuracy corresponds to one network environment and, accordingly, one started process, and thus a layer, the environment and processes can be identified under number of port. Start of necessary processes for the given example can be executed under the instructions resulted in *the Example 2* .

As well as elements of an operative computer memory, a condition of neuron can be interpreted by various images. Binary units of data, such as and байты, it is possible to interpret words as the letter, numbers or commands, and already on the basis of their combinations more complex structures are designed. The neuron, being the dynamic object capable not only to store the certain values, but also to carry out the actions connected with it, essentially expands opportunities of interpretation. For example, we can interpret a condition of group of neurons as model of the conceptual device of the brain which is responsible for primary perception of words or other signals, acting of touch groups. Otherwise we can consider neural excitation as indicators economic or productions - a condition of stocks of a warehouse, loading of the equipment, financial risks and so forth Thus integrated characteristics, interference and dynamic properties of neural models allow to simplify essentially data presentation, both for decision-making in real time, and for application of the mathematical device, for example, as methods of optimization, forecasting, classification, etc.

Dynamic properties of neural models allow simplifying the decision of the problems connected with the linguistic analysis. We shall consider an example in which we shall use group of the interconnected neural layers for the decision of a problem of the elementary translation of words from the several dictionaries representing various languages through an intermediate layer of equivalent concepts.

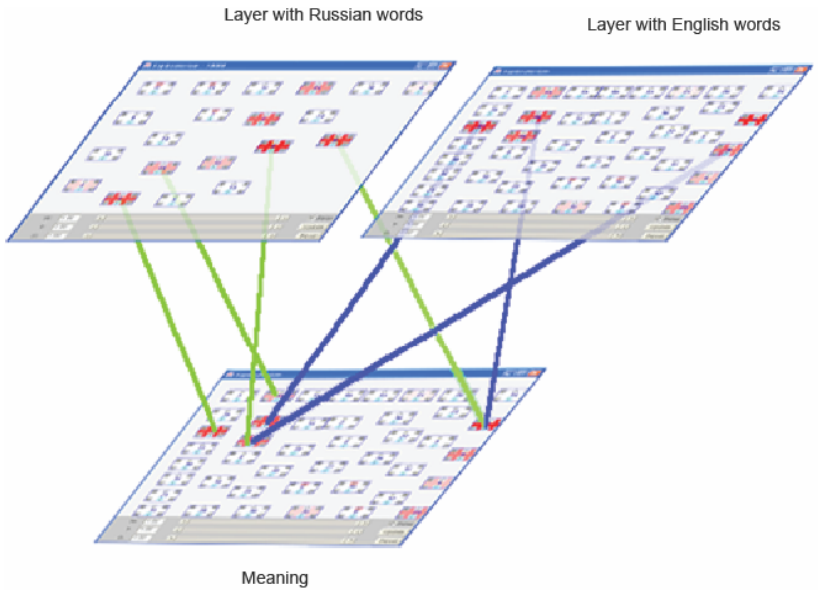


Fig. 39 Links between words and concepts
(Example 3 http://www.nnod.com/np)

We shall assume, that *10000* we shall use *a layer* for storage of the words belonging Russian (*dictionary* **RU**), and *a layer 100001*, accordingly, for words of English language (*dictionary* **EN**). We shall enter the third layer *10002* in which we can set semantic word meanings, or *concepts (dictionary* **M***)*. We shall admit, that to each word from dictionaries **RU** and **EN** there corresponds a unique word in dictionary **M** then we can construct the elementary multilayered neural network connecting words and concepts (see Fig. 39). Use of additional layer **M** allows

passing in the further to more universal concept - *sense (meaning)* when representation of a word is necessary for presenting in a kind independent of concrete language. For example, neuron $\mathbf{N_1}$ in layer **M** can represent a semantic word meaning *GENERATES* from dictionary **RU** and word *GENERATE* from dictionary **EN**.

Use of the semantic values synchronized with an entrance stream of words in alive speech, expands opportunities of linguistic analyzers which it is possible to build in such mechanism. In this example the stream of words from several offers which are checked on an accessory to the dictionary is used, are indexed, and then the program sends a signal of excitation of corresponding neuron. On Fig. 40 the scheme of interaction between three layers and the program of allocation of words is shown. Here it is supposed, that conformity between words is already established, and weights of this conformity are identical to all words.

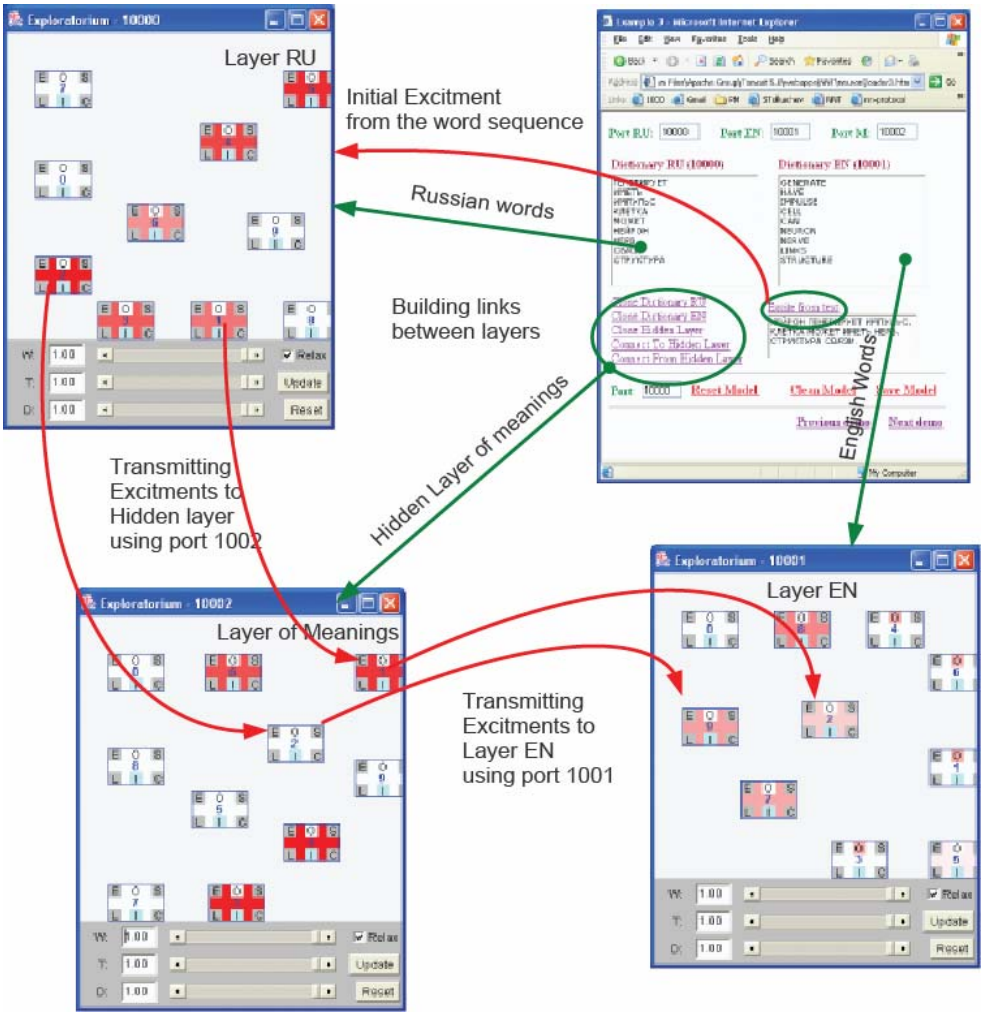


Fig. 40 Words, concepts connected to the Web page.
(Example 3 http://www.nnod.com/np)

In the further we shall use plural attitudes between words, and thus weight factors of these attitudes will be defined depending on their importance and contexts.

On this example it is possible to track all the basic features of the multilayered interconnected neural models. The excitation, developing on one of neural layers, can pass to other layer in which they can generate new streams возбуждений which in turn can pass to following layers and so on. The interconnected neural layers can incorporate in very complex structurally functional formations; therefore receptions of the conceptual analysis and decomposition are the important components of process of designing, allowing receiving a demanded level of detailed elaboration and representation. In some cases the requirements shown to presentation of representation, can essentially change appearance of neural model.

Words are elementary units of dialogue in human speech, and for each word there is a set of known values, as a rule, resulted in various dictionaries. Sequences of words acting in neural model, raising corresponding neural groups, lead to change of dynamic space of concepts of separate words and the contextual values formed during all history of development of concrete neural model. In such neural model the understanding can be defined as process of a finding of the optimal reaction in dynamic space of words and contexts. External display of understanding is always shown in the form of reaction, therefore a level of knowledge of system it is possible, from the point of view of the external observer, to reduce to adequacy of set of every possible reaction.

Just as for each separate word there is a set of dictionary values, and for various word-combinations there are various meta-values, and joint set of the most probable values defines a current condition of system - understanding of a current phrase which in turn, allows receiving the reaction most suitable for this understanding.

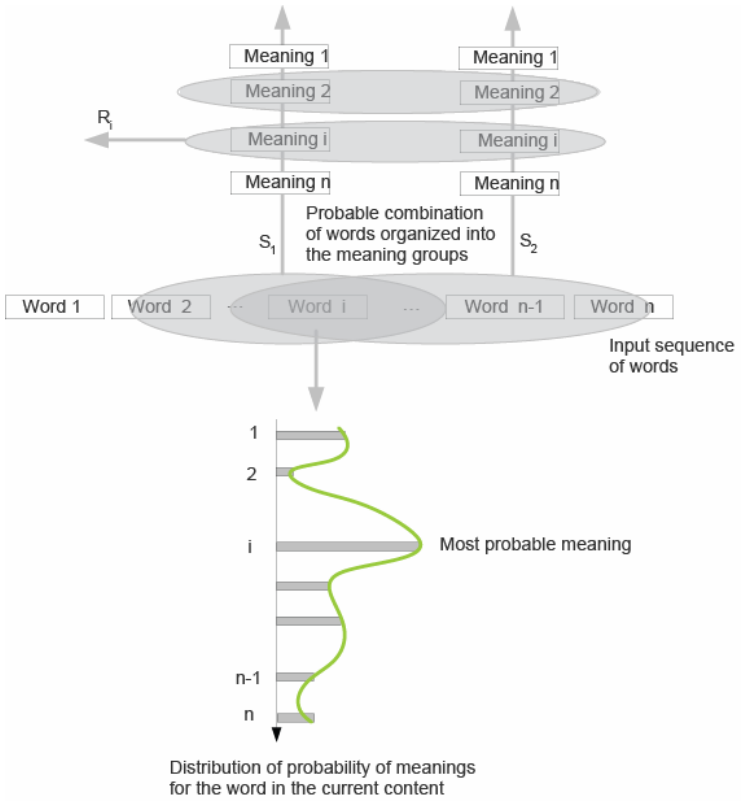


Fig. 41 Integration of specific word's meaning and combination's of words meaning

It is obvious, that depth of the analysis of dictionary values, as well as values of word-combinations, depends on capacity of neural model. Than the quantity of the various semantic values connected with various word-combinations more than variants of values in the dictionary of system and the more, the understanding is deeper and the more so reaction of system (Fig. 41) will be qualitative. The establishment of communications between word meanings and word-combinations is connected with necessity to have simultaneous access to a plenty of various information sections of neural model. And to construction of the user graphic interfaces intended for editing of systems of representation of knowledge, the big attention especially last years in connection with development of multimedia methods of representation of graphic data is paid to visualization.

On Fig. 42 the example of neural editor *NN Editor* (author A. Zaborsky) developed in *company NeMo (http://kiberry.ru)* for modeling of dialogues with virtual agents, placed on a web-pages is shown. The principle of gravitation of the representations [26] is put in its basis, allowing allocating from a plenty of the interconnected neural structures most close to current user representation.

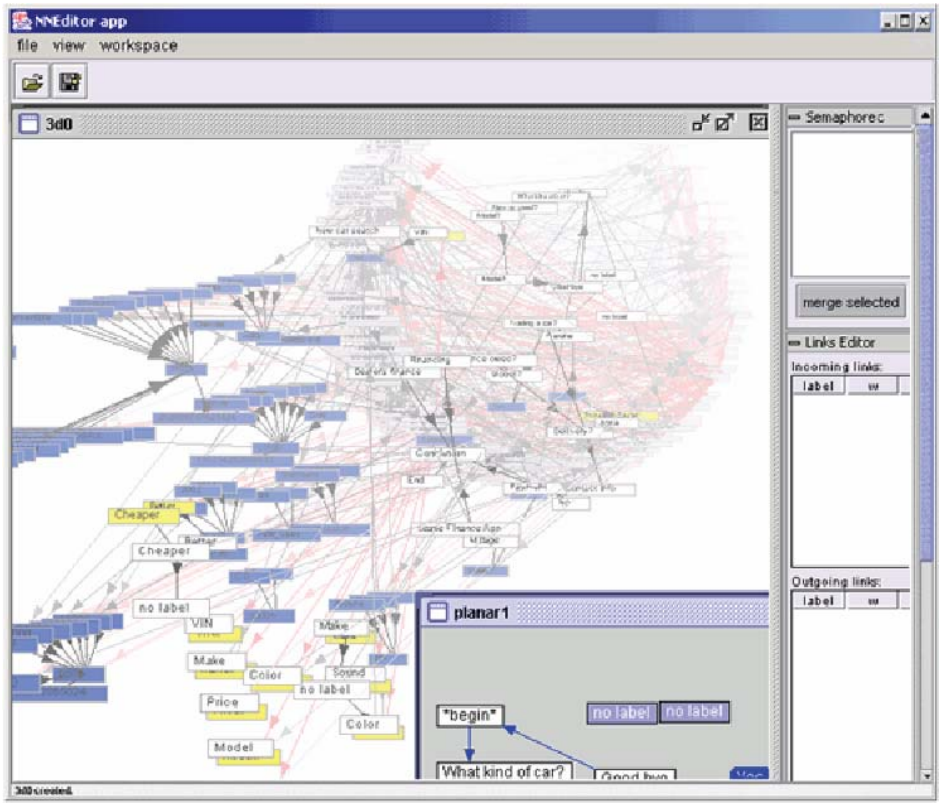


Fig. 42 The gravitational editor of neural models

PERCEPTUAL LINGUISTICS

In the middle of the last century in the physicist and the mathematician theoretical representations about borders of opportunities of reception of exact or absolute knowledge were generated at carrying out of physical supervision and measurements of properties and characteristics of objects of the real world or in logic constructions and the proofs proving the validity or completeness of any theories. Besides cleanly theoretical results which are of interest first of all for physicists and mathematicians, in them universal methodological positions which can be used at the decision of applied engineering, administrative or economic problems contain. Moreover, these results allow understanding in a new fashion the certain phenomena in areas of art, apparently, very far from formal sciences - poetry and painting.

The principle of uncertainty Heisenberg's [12, 13] in the physicist says, that it is impossible to define simultaneously precisely for a particle position **q** and an impulse **p**. The more precisely we define its position, the less we know about an impulse and on the contrary. Heisenberg distributes these restrictions on the basic dynamic variables describing physical systems - time and spatial coordinates **q** and **t**; and also an impulse and energy **p** and **E**:

$$\Delta p \, \Delta q \sim h, \qquad \Delta t \, \Delta E \sim h$$

where **h** - Planck's constant.

It is possible to consider, that the methodological sense of principle Heisenberg consists that at designing systems we should consider constantly effect of the interconnected parameters that at increase in accuracy of definition of one parameter to not lose value of another.

In the mathematician in 1931 Kurt Gödel [11] has proved theorems *of consistency and incompleteness* which if to formulate them informally, say, that any consistent axiomatic formalizations contain independent statements which it is impossible neither to prove in the theory of integers, nor to deny within the limits of the given formalization and if formalization in the theory of integers, is consistent it is incomplete. In programming theorem Gödel prove to be true quite concrete practical knowledge of that now there are no ways of the proof of a correctness and productivity of real programs, differently as through their performance. If to consider, that these theorems are true with reference to methods of construction of formal systems of the grammatical analysis of natural languages in mathematical linguistics, we never can construct the full and consistent linguistic system capable in full to understand and interpret alive speech on the basis of those principles which are used in their bases.

It is possible to find set of acknowledgement of validity of these restrictions in programming. The law of the Brooks that labor input by development of programs increases in 10 times in process of transformation of the simple program into system software product, reflect that fact, that the aspiration to receive the absolute and completed product unexpectedly leads to essential expenses at, apparently, final stage of integration. The Brooks, analyzing the isolated program systems on an example of operational system IBM OS/360, in the sixtieth years has shown, that there is no such software product as which it would be possible to consider completed - constantly changing environment leads to necessity constantly to add or modify properties of programs.

Principle of uncertainty Heisenberg and theorems Gödel form a constructive methodological basis, using which designers can define architectural restrictions and concentrate on the

directions, allowing receiving practical results optimum by. In problems of interactive interaction value of such restrictions is difficult for overestimating. Really, on the one hand, there is a constant aspiration to increase of accuracy of definition of word meanings and concepts of classical linguistics (linguistics), and with another - similar aspiration to creation of the full formal theory of languages within the limits of which it would be possible to define semantics and to the pragmatist of process of dialogue in mathematical linguistics.

At designing interactive systems with application of various forms of the organization of dialogue the programmer equally can be necessary knowledge of formal methods of construction and use grammars, no less than phonology, lexicon and semantics from the theory of linguistics. Thus for the programmer, as well as for the applied expert in traditional engineering areas, essentially important condition of the successful decision is the finding *of "golden mean"* among set various, frequently inconsistent theoretical and applied knowledge. The linguistics and mathematical linguistics in the certain degree remind physics and the mathematician, therefore analogies between principle Heisenberg and theorems Gödel are necessary to us that at designing applied interactive systems we could use reasonable restrictions and optimum combinations of results of researches in linguistics together with formal methods of the analysis and synthesis from mathematical linguistics.

So by means of various dictionary systems (for example, WorldNet [9]) we can create deep enough definitions and specifications of word meanings. Thus, dictionary systems represent only static characteristics of words or word forms in отрыве from their dynamic communications which can form rather complex context influencing complex perception and understanding during live dialogue. Principle Heisenberg with reference to the lexical analysis can be formulated as follows: the more precisely we define static value of a concrete word in a stream of live speech, the less we know about its contextual sense. The exact static word meaning in dynamic as cannot be defined environments, as well as to establish exact coordinates of a moving particle:

$$\Delta \mathbf{m}\ \Delta \mathbf{c} \sim \mathbf{l}$$

where

l - some linguistic constant (noise);
m - a static word meaning;
c - contextual sense.

Presence of the set forth above restrictions allows simplifying essentially modeling of interactive systems, however thus a choice and definition of concrete ways of modeling, degrees of formalization and depth of lexical representations entirely depend on the individual author. Here again it is possible to recollect, that one of the most fundamental works on programming refers to - "*The Art of Computer Programming*". The monograph «*Art of programming*» by Donald Knuth has rendered essential influence on formation of programming as scientific and technical discipline. The whip names programming - art and considers that programs can contain mathematical algorithms, to be optimized by means of mathematical methods, but thus concerns to the same works of art, as poetry or music.

Art possesses surprising ability to connect incompatible and to create the images transferring the information, the most unexpected way. It is not surprising, that such expressive ability involved interest of many mathematicians and representatives of other exact sciences. Among these researches for us special interest would be represented with works. Century Rauschenbusch on visual perception in painting [35, 36] in which it consistently analyzes perceptual both objective models of space and their possible representations in a brain of the person. Considering a problem

of the art image of the volumetric three-dimensional world on a plane of figure, Rauschenbusch comes to the unexpected conclusions which can appear rather useful at modeling linguistic processes.

The visual perception is based on the one hand, on well studied optical transformations to an eye, and with another - in it much less clear processes of processing of visions participate in a brain. In physiology, psychology, the physicist and the mathematician it is possible to find set of the various models concerning sight, however their association in one complete representation yes these times is rather a challenge. Mathematical models of visions are considered in geometry where depending on a choice of axioms it is possible to receive geometry Euclid, Lobachevski, Minkovski and others in which various theoretical and objective properties of a reality are considered. From psychology and physiology well-known the facts that internal representation and an objective real image can differ essentially from each other. The person restores three-dimensional space, using two-dimensional visual sensor controls. Distortions which are brought with a brain in visual representations, serve for more effective modeling a reality which being individual, depends on many conditions.

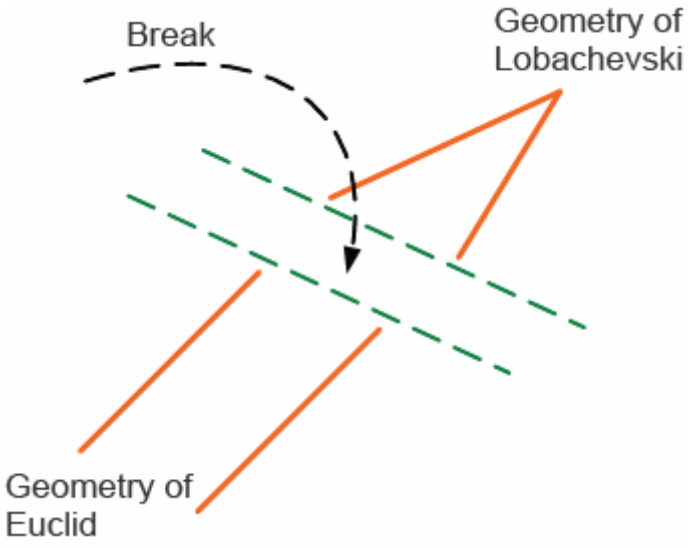


Fig. 43 Break between two geometrical forms in painting

If the visual perception can be named a direct problem artists solve a return problem of display on two-dimensional surface of a picture of some events or properties of an external world which generally, besides three-dimensional geometrical properties, can include dynamics of movement, time and a lot of other factors which set creates *an artistic image*. Considering process of creation of art images, Rauschenbusch analyzes properties objective and perceptual spaces and comes to the interesting conclusion: as geometrical properties of real objects and them perceptual images essentially differ, artists should find constantly individual decisions for creation of illusion of unity of two various geometrical representations. Analyzing properties of forward and back plans of pictures, visual perception as a whole, Rauschenbusch shows presence of breaks between two various geometrical models (Fig. 43), which artists, deliberately or intuitively, connect, solving thus a problem of overlapping of two геометрий - Euclid and Lobachevski.

On representations Rauschenbusch there are two basic ways of transfer of objective space - the drawing transferring objective geometry, and the figure transferring geometry perceptual. Thus,

the same subject can be represented in two ways - in the drawing both in figure, and thus both images will be true, as artists, at all times experimenting with their various combinations used.

During many centuries artists, solving extremely challenges of modeling of multivariate images, show a celebration of common sense above unilateral knowledge. Understanding a paradoxically of the certain forms and methods, artists find the various artificial receptions, allowing to eliminate breaks and to coct incompatible parts in the harmonious whole. Art of programming in this sense can be compared to art of painting to the full. In information systems it is necessary to overcome breaks and to combine methods and approaches in the same degree inconsistent and it would seem, incompatible as geometry Euclid and Lobachevski in painting. Thus programmers should solve practically equivalent problem of display of complex multivariate dynamic processes in the information representations possessing lower dimension.

Let's enter into structure of our information system perceptual space (Fig. 44) in which modeling representations and knowledge of an external world will be carried out. For creation of the complete system capable to intelligent reaction, it will be necessary for us to unite structural components of mathematical linguistics, dictionaries and thesauruses, classes and methods of object-oriented programming together with neural models.

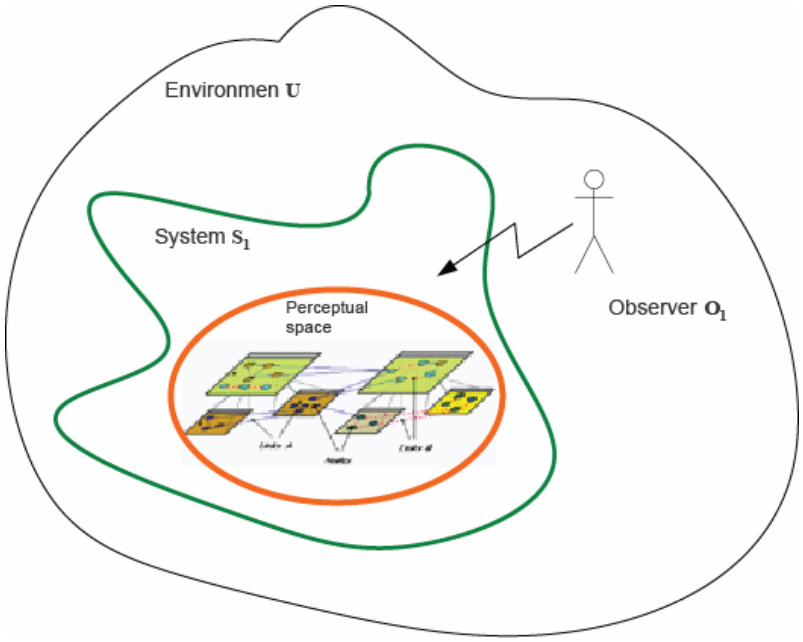


Fig. 44 Perceptual space in interactive system

Modern methods of mathematical linguistics with sufficient efficiency allow analyzing and synthesizing rather short fragments of speech. On the other hand neural models and semantic networks allow creating and supporting deep enough contextual representations. We shall understand under perceptual linguistics association of methods of mathematical linguistics and the neural models, capable to create and support perceptual spaces in information systems.

Just as projections of two-dimensional visions allow to recreate the three-dimensional image, and drawing signs and additional details (prospect, color, etc.) allow to recreate a full picture of real object, neural projections and the parametrical tables connected with them and procedures of reaction allow to receive representation about multivariate properties of information objects, events and processes and their reactions.

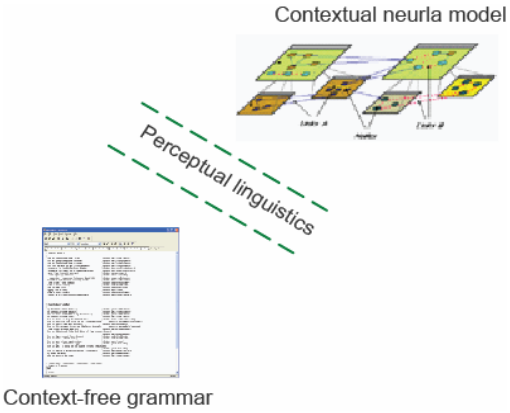


Fig. 45 Perceptual linguistics – the glue between context-dependent and context-free grammars

The practical embodiment perceptual linguistics in our case represents the intersystem interface, allowing to unify a stream of messages/reactions and to unite contextually-independent grammar with a dynamic context of neural models. For example, we can use grammar of recognition of speech for the analysis of a stream of voice messages and by means of neural models dynamically to expand them depending on a concrete context during dialogue.

CONTEXTS AND TEMPORAL LOGIC

In practice of programming well-known examples of successful application of neural models at the decision of problems of classification and recognition of images. However artificial neurons can be not less effective are used for construction of wider class of dynamic systems, including interactive and dialogue systems. As already it was marked above, the exchange of messages during dialogue leads to change of inwardness of systems participating in this exchange. Thus changes can occur not only in the system receiving the messages, but also in system which initiates these messages. Such changes, in their all set, form *a context* of concrete interaction. Generally the any quantity of systems can participate in this process; however we shall be limited to consideration of a case of an exchange by messages between two systems (Fig. 46).

It is obvious, that synthesis and the analysis of messages in these systems should occur depending on a context which, in turn, can be certain by a set of corresponding parameters. In real systems the context represents complex dynamic structure. It can include the information on various characteristics and properties of a discussed subject domain, and also on the attitude to them of the parties participating in an exchange. As examples of a context time, a place, model, cost, as much as linguistic properties of the message, such, for example, as an accessory to this or that linguistic group - a question, the statement, a greeting, etc.

Modeling of the contexts reflecting sense of current conversation is one of the key factors defining efficiency of interactive process. We shall consider some features of various methods of modeling which can appear useful at practical construction of interactive systems.

Logic modeling. At logic modeling contexts, for example, in the environment of programming languages, the problem finally is reduced to a finding of set of the logic functions including variables, interactions describing a context. At change of values of these variables, as a result of calculation of logic functions this or that procedure P_i which is responsible for reaction of the corresponding condition of system can be executed. If to assume, that exists n the variables defining a context of system S_i then the problem of a call of procedures P_i can be reduced to a finding and the subsequent decision of logic functions:

if $(f_1 (X_1, X_2, \ldots X_n))$ *then* P_1
if $(f_2 (X_1, X_2, \ldots X_n))$ *then* P_2
...
if $(f_m (X_1, X_2, \ldots X_n))$ *then* P_m

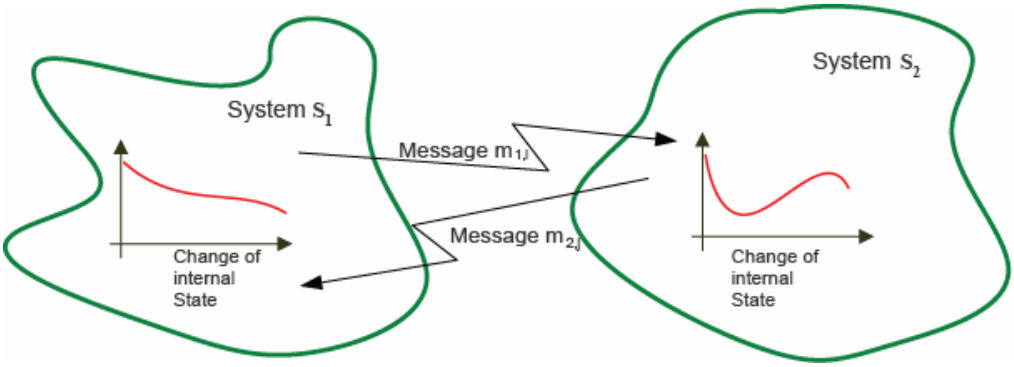


Fig. 46 Interactive communication between two systems

At use indistinct and time logic for the comparative small and limited set of variables it is possible to receive quite comprehensible decisions. However at increase in quantity of variables and interrelations between them there is an avalanche increase in complexity therefore the common decision on the basis of logic modeling appears practically impossible. In real systems the quantity of the variables influencing a context, can total some thousand, besides, that the quantity of communications between them can be the same order. The most complex at programming such logic functions is the open character of systems that assumes constant change, both quantities of internal variables, and interrelations between them.

Hardware modeling. At use of hardware models, at the disposal of designers are available various functional and logic, in that count also analog, elements - memory, decoders, delays, converters and т. Item in which basis of work the consecutive logic lays. At designing hardware systems by one of the most critical factors time is. Distribution of signals, logic of operation of functional elements, transients at switching - all this has the certain time characteristics which account leads to necessity of application of synchronization, gating, delays and other ways, allowing to receive logic decisions in conditions of real time.

Distinctions between the hardware and logic modeling, connected with the factor of real time, have fundamental character. Events in hardware systems have the natural asynchronous nature, unlike logic modeling in program environments where practically always there is an opportunity to order streams of events by means of subsystems of planning and management of turns. The analysis of time diagrams with the purpose of their synchronization is one of the basic methods of designing in hardware modeling. For this purpose during modeling combine logic schemes and time diagrams. On Fig. 47 the example of such overlapping where as a consecutive logic element the elementary binary counter is chosen is resulted.

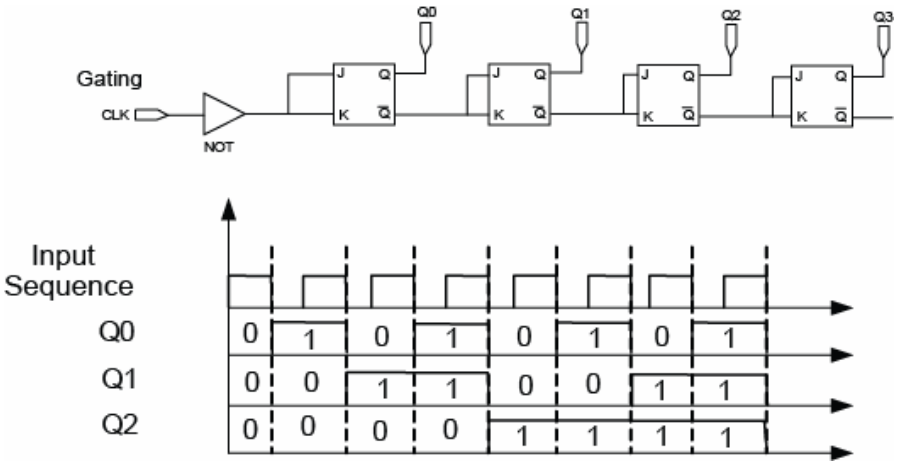


Fig. 47 Example of the binary counter and its time diagram

There is one more area of human activity where formal notations and definitely connected sequences of executions are synchronized by the closest image in real time. It is a question of music where original language formalization of musical records, and if so it is possible to be expressed is used, the technology of designing of pieces of music, in general, is very similar to technology of hardware-software designing of logic devices. The role of logic in music is executed with harmony in which the axiomatic bases for laws of association of sounds and accords in sequence are defined, within the limits of this or that musical theory.

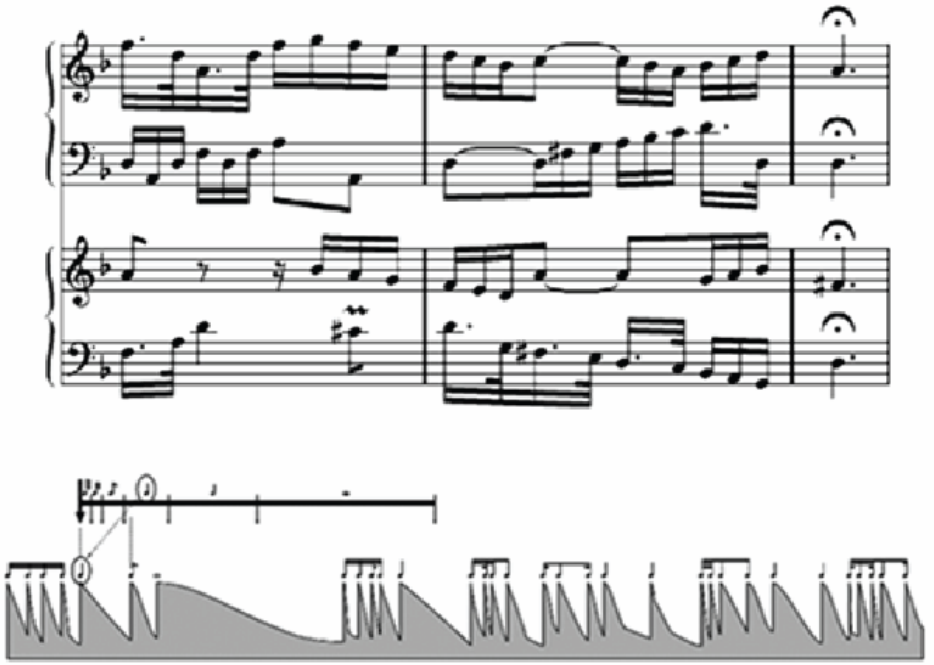


Fig. 48 Time programming in music
Copyright (c) 1999 James Ingram
(http://www.the-notation-of-time.de/Website1999-2004/devmus.htm)

As well as in mathematical logic, in music there are the of axiomatic, languages, theories of construction and interpretation. If to consider musical notations from the point of view of formal languages it is possible to consider, that its alphabet consists of set of notes of the certain frequency and duration. Unlike classical operation concatenation which is defined in the theory of formal languages as operation of connection of symbols in sequence, in musical notations конкатенация is multivariate, that is notes can be connected in chains in many directions. As a whole pieces of music are similar to programs and their language designs can be interpreted in parallel and asynchronously.

Ways of the description of the parallel and asynchronous executions, applied in music, represent special interest for neural modeling. On Fig. 48 the example of a piece of music and the time diagram of sequence of notes in the form of peak-time characteristics from James Ingram's work "*The Notation of Time*" [14] is shown. Execution of pieces of music can be considered as set of parallel processes developing time, порожденных on the basis of initial musical texts. Each of these processes consists of sequence возбуждений and braking of some musical objects which possess as a whole rather small dynamic set of well formalizability acoustic properties. If to consider, that each such process consistently raises and then definitely changes amplitude of a sound according to the partita, we can present musical execution in the form of three-dimensional space of processes as it is shown on Fig. 49.

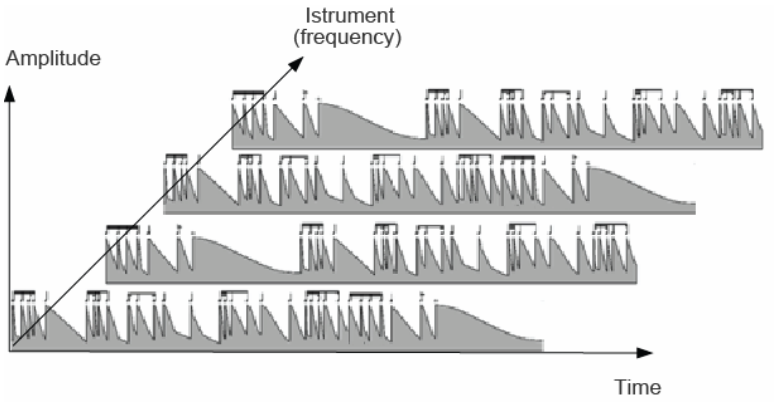
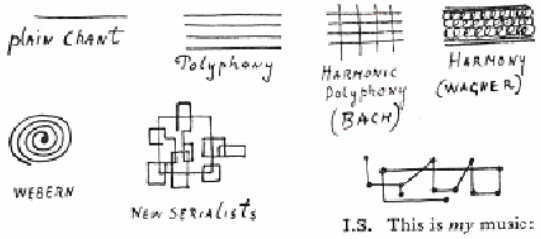


Fig. 49 Peak-frequency representation of execution of a piece of music

Here it will be pertinent to result as an example I. Stravinsky's figure from its book *"Conversations With Igor Stravinsky"* [25], describing its representation about harmony and ways of development of musical themes as the processes developing in time and space. Stravinsky's representations about various forms of construction of pieces of music can be used at the organization of dialogue between the person and information system. Leaning on its representations about dynamics of development of pieces of music, it is possible to receive set of various architectural decisions of the organization of dialogues - the simple linear conversations, the parallel linear not crossed themes, the linearly-connected themes, etc.



In notes and their musical interpretations discrete, is formal-logic descriptions which the author and sets the analog parallel execution, are combined by musicians interpreting these partitas. If to compare Fig. 47 where the peak-frequency characteristic of musical execution and Fig. 24 on which development возбуждений in neurons it is possible to see is shown is presented, how much these two models are similar. Neurons possess properties very close to notes and characteristics, and their ability to excitation and braking allows speaking about an opportunity of application of some of formal properties of musical models in neural programming.

Rather simple musical language allow to create a unusual variety of musical compositions - from simple melodies before extremely complex symphonic products, thus in it is possible to allocate some characteristic differences from properties of programming languages.

1. The musical alphabet includes the elements possessing time properties.
2. Concatenation of elements of the alphabet in chains can have some directions.
3. The space where execution of musical records is carried out is multivariate.
4. The form of record (musical stave) possesses an opportunity to create a plenty of parallel processes and to synchronize their execution.

In musical language, as well as any the friend, it is possible to create the designs possessing certain properties, and to spend their analysis, applying various formal or informal methods. For

neural models there should be not less effective way of the description, time logic and language in which it is possible to create time designs, applying which it is possible to spend the analysis and synthesis of neural structures. Actual similarity of neural and musical models allows taking advantage of some musical receptions and methods, with reference to formal methods of the description and interpretation of neural systems.

In comparison with musical systems, neural models possess higher level of dynamism - in neural models *"score"* can develop and modify the properties in time. Ability of neurons to cloning and a dynamic establishment of communications with other neurons allow speaking about programming programs in full sense of this concept.

Let's consider as an example cloning of some basic typical neural structures. In computer facilities as the elementary remembering element the trigger - the device from two base logic elements is used, capable to keep one of **n** steady conditions and to be switched depending on an entrance signal. The feedback connecting in pairs outputs of logic elements are applied to construction of the trigger, with the inputs which are responsible for dump of the given element in a zero condition. On Fig. 50 are shown the elementary binary *R-S (Set-Reset)* the trigger and its binary neural equivalent.

Four neurons connected in pairs will be necessary for construction of a neural equivalent of the binary trigger, and direct communications have weights: $\mathbf{w_{0,1}}$ and $\mathbf{w_{2,3}}$ are equal **+1**, and the return - $\mathbf{w_{0,3}}$ and $\mathbf{w_{2,1}}$ are equal**-1**. At such connection, in case of excitation of neuron **0** that corresponds to an individual signal on input $\mathbf{I_1}$ (it is supposed, that threshold value of neuron equally or there is less than level of entrance excitation), neuron **0** will distribute excitation through all the outputs. Thus neuron **1** will receive a positive individual entrance signal which will translate it in the raised condition, and neuron **3** will receive a negative individual signal which will dump its excitation if this neuron has been raised before it. Neuron **2** in the same way works. Thus, this scheme completely repeats work of the trigger.

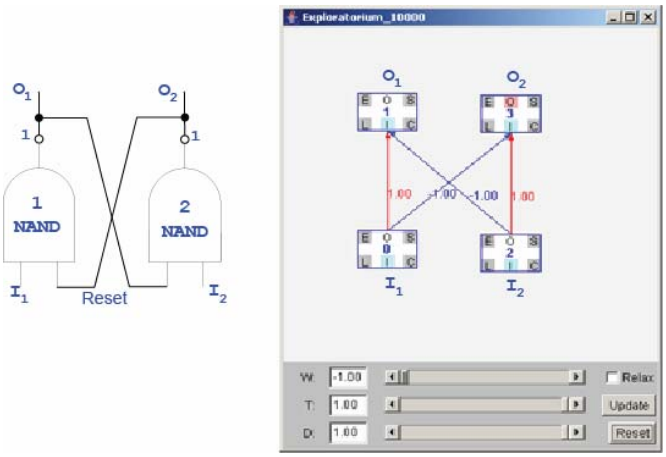


Fig. 50 The electronic trigger and its neural analogue

Unlike an abstract logic element, the neural trigger has additional parameter - time of switching, operating which it is possible to receive various time delays. Having closed pair triggers in a ring, it is possible to receive a network loop - model of an infinite cycle. Interpretation of loops, or infinite cycles, in neural models have practical sense. As at them there is time, the infinite cycle can be used as the synchronizer of various processes and events as the majority of operations in

neural programming depends on time, unlike the operator in programming languages where the majority of operations should be invariant by time.

Let's consider more in detail process of creation neural a component. The binary neural trigger it is easy to expand and receive multi-port remembering element for **n** conditions. On Fig. 51 the decision for the neural trigger having six conditions is shown. Using , we can construct easily enough the binary neural trigger, cloning proto-neuron and consistently connecting pairs neurons in the received group, setting weights of communications accordingly **+1** and**-1**. However if we shall want to construct the neural trigger having 100 conditions, hardly it is possible to consider manual programming expedient. At first sight can seem, that this problem simply enough to solve traditional program methods. Really, if there is a program-emulator in which neural models it is obvious, that inside of this program are supported, probably to create the subroutine or a method, capable to construct such trigger, for example as follows:

```
Number-of-States = 100;
for (i := 1; i <= Number-of-States*2; i++)
{
        make-New-Neuron(i);
}
for (i := 1; i <= Number-of-States; i++)
{
        for (j := Number-of-States+1; j <= Number-of-States*2; i++)
        {
                w = -1;
                if (j == i + Number-of-States) w = 1;
                make-New-Nerve(i, j, w);
        }
}
```

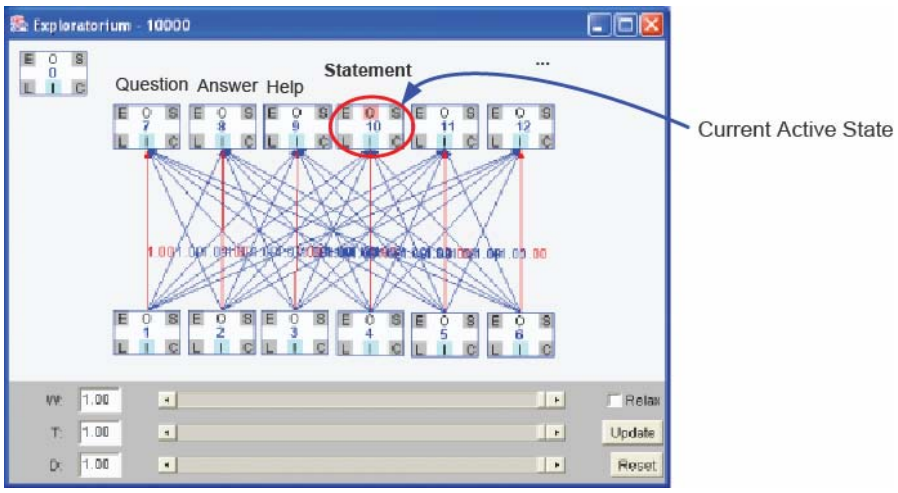


Fig. 51 System from six neuron triggers

Performance of such program probably or directly in a body of a neural environment, or is required to us the interface similar on SQL/ODBC. In any case if programming of regular neural structures can be reduced to simple enough programs, complex irregular structures will lead to essential complication of algorithms and actually programming. Using the mechanism of cloning and time delays, the same decision can be reached by a different way.

If to connect together time logic of cloning and harmonious laws of music programmers could create the sequences operating development of neural systems, and thus quality check of the decision, debugging and revealing of mistakes could be carried out in the same form as it is done by the composer and the conductor.

It is possible to assume, such process of programming would be how much more aesthetic!

CLONING

Ability of neurons to cloning opens opportunities for creation of complex multi-coherent structures by analogy to how it occurs to synthesis of fibers in biological organisms. Before passing to the analysis of program methods of cloning, we shall consider as there is this process in biological systems. It is possible to carry to the most outstanding in histories of mankind opening Дж. Watson and F. Crick [29] double spiral structures of salt of a deoxyribonucleic acid *(DNA)* which underlies reproduction of biological cells. On Fig. 52 it is shown structural (*)*, molecular (*)* and information (*)* representations of this molecule. What complex were structure and the organization of a circuit *of DNA*, from the point of view of programming it represents sequence of codes which can be interpreted by the executive mechanism - the processor which role in a cell is carried out with ribosome.

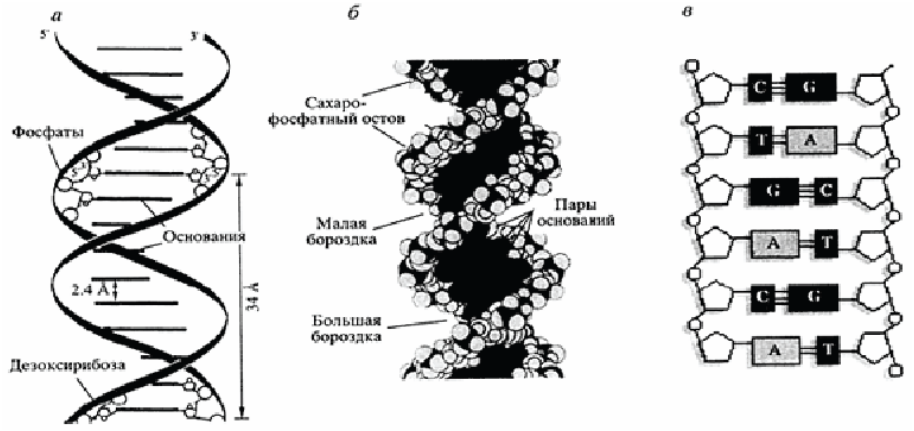


Fig. 52 Three representations of DNA

If to consider, that the ribosome is an executive processor and molecule *RNA* - the program synthesis of fibers is anything to others as parallel execution of sequence of the codes which have been written down on the carrier - tape *PHK* received by copying of a part of other carrier - tapes *of DNA*. All other details connected with process of direct chemical synthesis, for programmers do not represent special interest as well as actually the nature of execution of commands in any processor - be it a biological ribosome or silicon *Intel 8080*.

On Fig. 53 process of synthesis of fibers of sequences of three nucleotide - codon, corresponding amino acids responsible for connection of which eventually and fibers are formed is shown. кодонов it is possible to consider quantity commensurable with quantity of codes in complex programs, such for example, as operational system. Unique difference between the modern processor and cellular processes is a speed of interpretation. It is possible to consider, that in a basis of cell fission process of interpretation enough the simple codes forming very long sequences lays. Thus in biological systems the important role external factors - play enzymes, electromagnetic fields and others which can be presented as the additional parameters influencing execution.

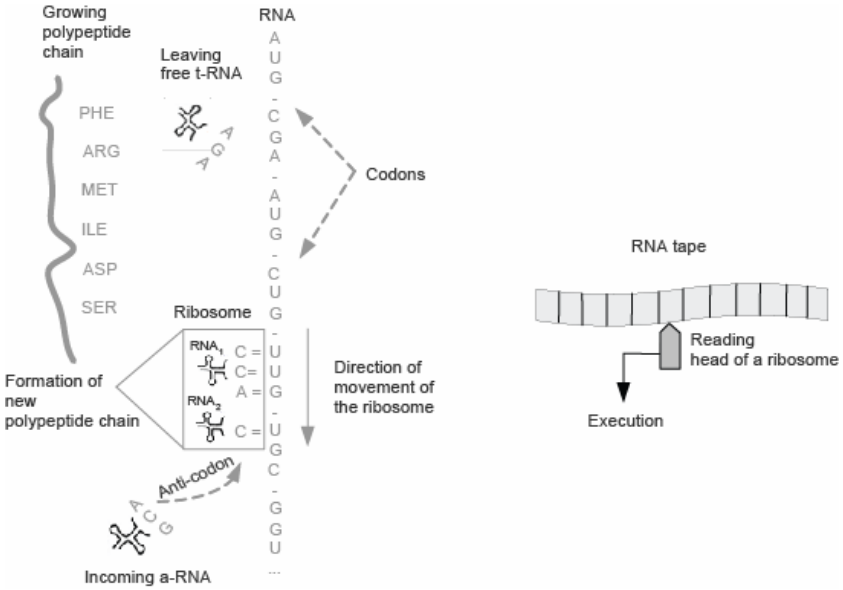


Fig. 53 The biological scheme of synthesis of fiber and its program representation
Copyright (c) the Greater Soviet Encyclopedia (Translation in biology)
(http://encycl.yandex.ru/art.xml?art=bse/00080/15000.htm&encpage=bse)

If to compare process of synthesis of fiber in a cell and computing system Turing [28], it is possible to see, that between them there are many general structurally functional properties. System Turing (see Fig. 54) includes the computer and a tape consisting of set of cells in which symbols can be written down. During each moment of time the computer processes one cell of a tape containing one symbol - **S(i)** and is characterized by a condition $\mathbf{q_i}$ from final set of conditions $\mathbf{q_n}$. The behavior of the machine is defined by a set of commands which are defined by pair $\mathbf{q_i}$, **S(i)**. Depending on a configuration in which it is, the machine can shift a tape to the right or to the left, to write down or read out symbols in cells and to change the condition (configuration). This machine is the elementary programming system, which A. Turing used to specify first of all concept of algorithm and вычислимости. Despite of cleanly theoretical character of its work and primitiveness of base operations, we can use model of machine Turing for construction of the practical mechanism of cloning of artificial neurons.

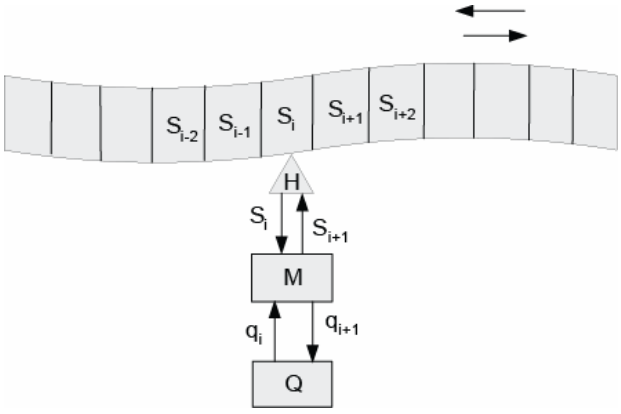


Fig. 54 The scheme of work of system Turing

Let's expand the internal mechanism of touch reaction of neuron, having added to each sensor control the entrance turn, capable to store inside of themselves sequences of time delays which will define time after which, the given sensor control of neuron will be raised (Fig. 55). We shall represent entrance signals as pair - time of a delay and amplitude **(t_i, a_i)**. Thus, the amplitude of an entrance signal should be нормирована from**-1** up to **+1**, and time of a delay should be set in the form of an integer in milliseconds. For example, we can load into the shop connected with a sensor control of excitation, sequence (*10 000, 0.5*), (*5000, 1*), (*2000, 1*). In this case neuron will be excited consistently three times, in *10* seconds with a level of excitation *0.5*, then in *5* seconds with a level of excitation *1* and in *2* seconds with a level of excitation-*1*.

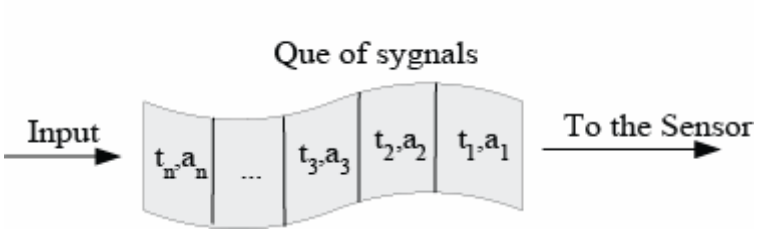


Fig. 55 Turn of entrance signals

Addition in model of neuron of turns of the entrance signals possessing time characteristics is obvious, that, assumes presence of the mechanism of internal and external synchronization. Synchronization is one of fundamental properties of real systems, whether it is biological, social or technical devices. The turns of signals connected with sensor controls inside of neuron, allow passing to construction more the general model of cloning capable to reproduction of complex structures, on the basis of rather simple sequences of codes which can be presented in the form of a tape, by analogy to the biological and mathematical tapes considered above.

Let's admit, that during the moment of excitation of a sensor control *the Clone* (see Fig. 33) neuron before to execute operation of cloning, will try to find a tape *suitable* for it which we shall name *Neuro-DNA* and if such tape exists to execute cloning according to sequence of the codes which have been written down on this tape. Under such tape for neuron we shall use a file which can be in the certain place and to have the name: **i.dna**, where **i** - number of neuron. Below the fragment of a method of cloning of neuron where it is shown how there is a reading a file to data for cloning is resulted:

```
try
{
        String file-Name = this.id + ".dna";
        FileInputStream fstream = null;
        fstream = new FileInputStream(file-Name);
        BufferedReader in
        = new BufferedReader(new InputStreamReader(fstream));
        String theInput = "";
        while ((theInput = in.readLine()) != null)
        {
                ...
                ...
                this.cloneNode(v-S-DELAY, v-L-DELAY, v-C-DELAY);
```

```
        }
}catch (IOException e)
{
        this.cloneNode(null, null, null);
}
```

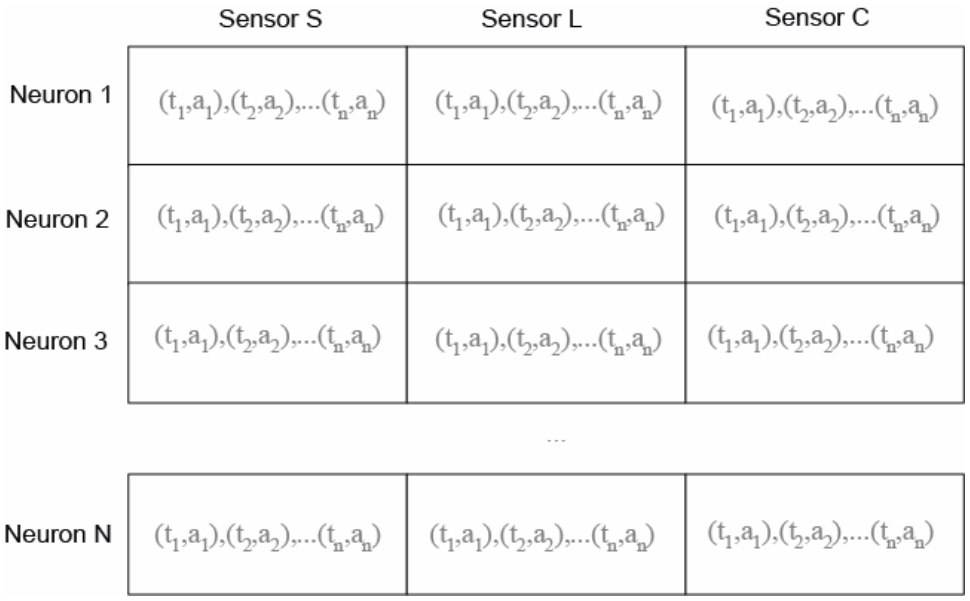


Fig. 56 Structure of file Neuro-DNA

The structure of a file represents sequence of lines (the sequence of symbols limited *CRLF* in *MS Windows*), each of which in turn consists of three sequences of the pairs divided *LF* (Fig. 56).

The algorithm of cloning of neurons is extremely simple: proto-neuron, initiated for performance of this operation, starts to read file *Neuro-DNA* belonging it, in process of reading the next line creates a clone and loads into its touch turns corresponding sequences of pairs delays and amplitudes of signals. *Neuro-DNA* it is possible to present also as the linear sequence of pairs divided on under-sequence two types of special symbols - markers. One of these markers - **NN** (*New Neuron*) initiates creation of new neuron. Other markers - **QSS**, **QLS** and **QCS** (*Queue to Select/Link/Clone Sensors*) define a sensor control in again created neuron in which turn the group of pairs following this marker will be written down.

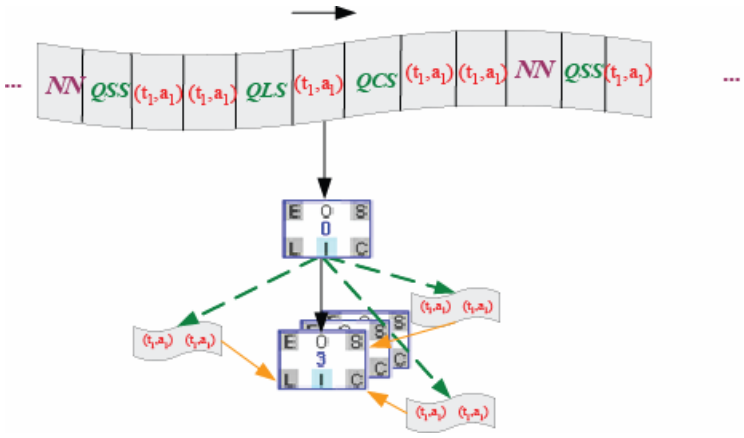


Fig. 57 Cloning of proto-neuron as a result of linear reading tape Neuro-DNA

Concrete example of file *Neuro-DNA* for proto-neuron **0**, allowing creating the neural trigger for six conditions (see Fig. 51), is resulted in table 3.

Таблица 3

Neuron	Sensor S	Sensor L	Sensor C
1	(0,0)	(1500,1),(5500,-1)	(0,0)
2	(0,0)	(10500,1),(5500,-1)	(0,0)
3	(0,0)	(20500,1),(5500,-1)	(0,0)
4	(0,0)	(30500,1),(5500,-1)	(0,0)
5	(0,0)	(40500,1),(5500,-1)	(0,0)
6	(0,0)	(50500,1),(5500,-1)	(0,0)
7	(1000,1),(14000,1),(10000,1),(10000,1),(10000,1),(10000,1)	(0,0)	(0,0)
8	(5000,1),(5000,1),(15000,1),(10000,1),(10000,1),(10000,1)	(0,0)	(0,0)
9	(5000,1),(10000,1),(5000,1),(15000,1),(10000,1),(10000,1)	(0,0)	(0,0)
10	(5000,1),(10000,1),(10000,1),(5000,1),(15000,1),(10000,1)	(0,0)	(0,0)
11	(5000,1),(10000,1),(10000,1),(10000,1),(5000,1),(15000,1)	(0,0)	(0,0)
12	(5000,1),(10000,1),(10000,1),(10000,1),(10000,1),(5000,1)	(0,0)	(0,0)

To fur-tree to present it in a tabulated kind in format *Excel* this file has 12 lines and 3 columns and according to it proto-neuron, carrying out operation of cloning, will construct 12 new neurons and will place in touch turns of each of them corresponding sequences of pairs delays and amplitudes. The process of cloning corresponding structure of communications between 12 neurons, is resulted in the Example 4. In this example if to excite sensor control **C** *(Clone)* neuron **0**, there will be a group between which elements communications according to time delays and the weights set in a file **0.dna** will be established.

Cloning and simultaneous creation of communications between neurons it is possible to consider as analogue of programming, however mechanisms which underlie neural processes, are identical both to creation of functional structures, and for their execution. In neural models at development and updating of internal properties the same mechanism of excitation is used, as at reaction to

external signals. If in computing systems the program and a computer have essentially various nature, in neural systems creation of functional structures and their interpretation are carried out by identical image. Thus the neural system is highly parallel formation, after duplication each neuron begins own development.

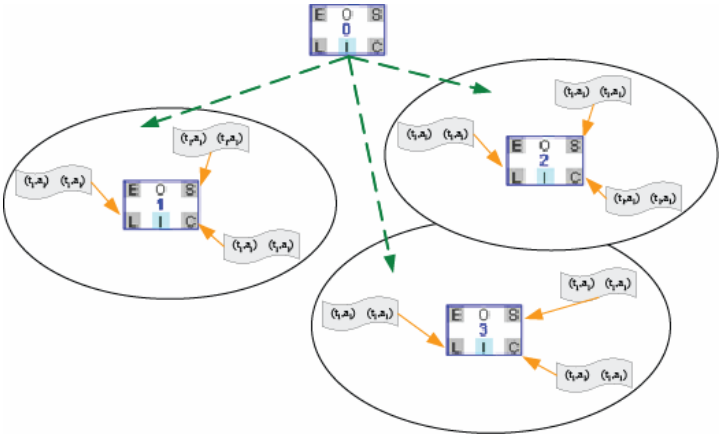


Fig. 58 Duplication of neurons with the set time parameters

On Fig. 59 the time diagram of formation of connections between the neurons forming the trigger for six conditions is shown. By red color are shown excitation as a result of which positive communications will be constructed, and blue - to sequence возбуждений which will lead to creation of negative communications. The period of a finding of neuron in a condition *"is chosen"*, in which neuron passes at excitation of sensor control **S**, here is accepted equal *5* seconds or *5000* milliseconds. In the given example relative time intervals are used, and for the coordination of possible losses during interpretation (similar to transients in real physical schemes) we shall use compensating time amendments which for the given example are equal *500* milliseconds.

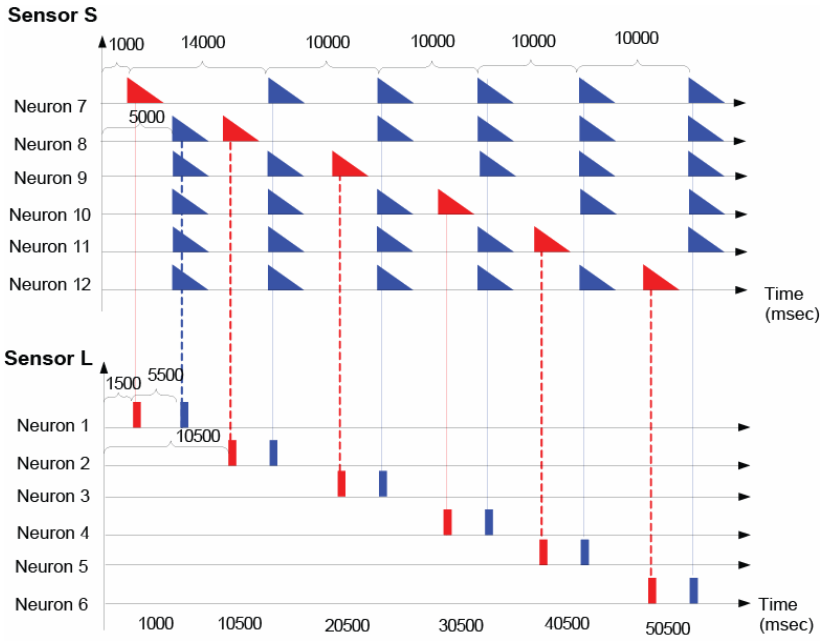


Fig. 59 The time diagram of construction of the neural trigger for six conditions

It is supposed, that time of cloning of all neurons negligible is not enough, and during some moment, all of them start internal processes of processing of turns of the events created proto-neuron during the moment of their creation. Through *1000* milliseconds after a birth neuron *7* will pass in a condition *"is chosen"*, in which it will be *5000* milliseconds. Through *1500* milliseconds, neuron *1* will excite sensor control **L** (*To construct communication*) and during this moment positive communication between neuron *1* and neuron *7* will be formed. Through *5000* milliseconds neurons *8-12* and through *7000* (*1500* + *5500*) milliseconds again excite neuron *1* therefore negative communications between neuron *1* and neurons *8-12* and so on until communications of this trigger will be formed all will be established will be excited.

The period of time during which neuron has the raised condition, can be chosen for a basis for definition of base frequency of synchronization in model as a whole. In the further we shall use minimally possible time during which neuron possesses ability to connection – τ as special size which depends on the concrete computing system and can be calculated empirically. For example, in our case τ = 10000 milliseconds.

Thus, the behavior of each neuron can be presented in the form of sequence of events which can be synchronized with other neurons both by means of relative time delays, and by introduction global gated sequences. We in the further shall use the combined methods of synchronization, whenever possible not complicating structure of artificial neuron needlessly.

SELECTIVE CONTEXT

What developed were not the form of dialogue between the person and interactive systems, without their information filling by subject data such systems have rather limited scope. Neural models allow to create the systems, capable to support complex dynamic contexts. An example of such systems is dialogue and search systems in which phrases or search inquiries substantially depend on a context. The greatest practical interest for modeling virtual representatives represents association of search methods with dialogue forms. We shall consider as an example conversation *of the Buyer* and *the Seller* in hypothetical virtual shop:

> ***The buyer***: *I would like to get the digital chamber.*
>
> ***The seller*****:** *Here, for example, some chambers, with various qualities. What of them is pleasant to you more?*
>
> ***The buyer*****:** *Here this chamber arranges me, but whether is not present at you similar for smaller cost?*
>
> ***The seller*****:** *Here some cheaper models.*
>
> ***The buyer*****:** *I would like similar, only from Sony.*
>
> ***The seller*****:** *Please, here these chambers are made by company Sony.*
>
> ***The buyer*****:** *It that me interests, but the chamber with the best increase is necessary to me.*
>
> ***The seller*****:** *At chamber Sony MVC of CD and the sanction fine, and record is carried out directly on a digital disk.*
>
> ***The seller*****:** *Yes, all is good, only a little bit heavy. Whether is not present similar, but it is better...*
>
> *etc.*

It is possible to find all in this example the basic problems with which developers of interactive help and commercial systems on the Internet collide. Than the buyer or the user, the more widely a spectrum of properties of production - the more difficultly process of a choice is more formed and informed, the cost of service is required to more time for the seller or the expert on service for satisfaction of inquiry and accordingly, the more expensively. Modern formal methods of the analysis and synthesis of natural languages not in a condition to offer the decisions allowing economically effectively to design similar dialogues, first of all because practically each phrase in them is context-dependent, on the one hand, and with another - data about production should be received from concrete, as a rule, a relational database for which inquiries should be formulated in the classical logic form.

Necessity to coct data from relational bases together with the linguistic designs depending on a context induces us to search for new synthetic methods of construction of interactive systems. We shall consider structure and dataflow in hypothetical system which is capable to carry on the above-stated dialogue.

Let's assume that the elementary structure of a database “Cameras” for digital chambers can be set by following tables:

Model	*- the name of the chamber (LS443, Easy Share DX4330 ...)*
Make	*- the manufacturer (Sony, Canon, Fuji ...)*
Price	*- the price ($400, $1200, ...)*
Description	*- the description*
ZOOM	*- factor of increase (2x, 4x, 12x ...)*
Matrix	*- a matrix (2MP, 3MP, 4MP ...)*
SCU	*- identification number*

It is possible to assume, that *the Seller* familiar with *SQL*, could, using the terminal of access to a database, during the above-stated conversation to send inquiries and to transfer *the Buyer* results. It is obvious, that for such type of structures of databases there can be rather a small number of variants of inquiries which can satisfy the majority of buyers if the concrete values of the variables describing a current context of given conversation are set. For example:

***SELECT * FROM** Cameras **WHERE** Price < $800 **AND** Price > $300*

***SELECT * FROM** Cameras **WHERE** Price < $800 **AND** Price > $300*
***AND** MAKE = ‘Sony’*

***SELECT * FROM** Cameras **WHERE** Price < $500 **AND** Price > $200*
***AND** MAKE = ‘Sony’ **AND** ZOOM > 3*

…

If *the Seller* will remember the manufacturer, a price range and other characteristics which interest *the Buyer*, it can always formulate following inquiry to a database. However to answer on such, for example, *whether* questions as *« Are not present similar, but it is better... »*, we should expand structure of a database and to add in it linguistic *(Fuzzy)* values [32].

Let's add in structure of our database for each variable which has numerical expression, it normalized representation. So, for example, for a variable *the Price (Price)*, we shall create normalized value *Price-Norm* which can be calculated as follows:

$$\textbf{Price-Norm}_i = (\textbf{Price}_i - \textbf{Min- Price}) / (\textbf{Max- Price} - \textbf{Min- Price})$$

Where
Price-Norm$_i$ - нормированное value (*0*> **Price-Norm**> *1*),
Price$_i$ - value of the price for i-th object,
Max-Price - a ceiling price among set of all prices,
Min-Price - a floor price among set of all prices.

Such values can be received for each change having numerical type, including date, cost, weight, etc. Example of a database in which are calculated normalized values for the several variables defining properties of digital chambers, is resulted in table 4.

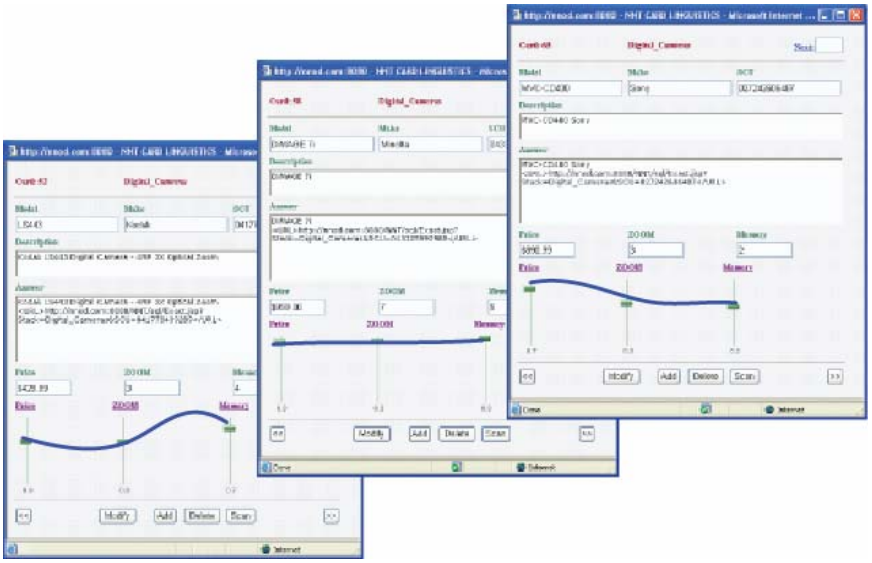


Fig. 60 Sequence of the cards containing the characteristics of digital chambers

Apparently from this example, we can use the dim values for such variables, as *the Price*, *Factor of increase*, *the Matrix* and т. Item On Fig. 60 the sequence of several records from a database *"Cameras"* in the form of cards is shown. The card represents the graphic interface to a database and contains the fields, allowing to display and modify records in a database. Besides the card allows to display нормированную the integrated characteristic of each concrete chamber. Thus, the card allows to connect together exact linguistic characteristics which can be set by keywords - *Sony*, *CD400* and the dim approached values, allowing to enter linguistic variables - *more*, *the average*, *is better* and т. Item for which areas in corresponding groups нормированных values can be set.

Table 4 A fragment of a database *"Cameras"*

Model	Make	Price	Mtrx	ZOOM	Price-N	ZOOM-N	Mem-N	SCU
LS443	Kodak	$429.99	4	3	0.31	0.29	0.69	41778439289
EasyShare DX4330	Kodak	$296.99	2	3	0.2	0.29	0.2	41771580506
EasyShare LS420	Kodak	$270.99	2.1	2	0.17	0.14	0.22	41778963470
EasyShare CX4230	Kodak	$198.49	2	3	0.11	0.29	0.2	41778843710
Pocket	Logitech	$95.49	1.3	1	0.02	0	0.02	97855014672
DiMAGE F100	Minolta	$417.49	4	3	0.3	0.29	0.69	43325992964
DiMAGE 7i	Minolta	$999.00	5	7	0.82	0.86	0.94	43325992988
DiMAGE S304	Minolta	$573.49	3.3	4	0.44	0.43	0.52	43325992247
DiMAGE 7Hi	Minolta	$1,196.99	5	7	1	0.86	0.94	43325993374
DiMAGE X	Minolta	$389.49	2.1	3	0.28	0.29	0.22	43325992780
Coolpix 4500	Nikon	$631.99	4	4	0.49	0.43	0.69	18208255030
Coolpix 5700	Nikon	$1,047.99	5	8	0.87	1	0.94	18208255047
Coolpix 885	Nikon	$469.49	3.2	3	0.35	0.29	0.5	18208255054
MVC-CD400	Sony	$890.99	2	3	0.73	0.29	0.2	27242606487
MVC-CD250	Sony	$595.99	2.1	3	0.46	0.29	0.22	27242606524
MVC-CD300	Sony	$858.99	3	3	0.7	0.29	0.45	27242589223
MVC-CD200	Sony	$578.99	2.1	3	0.45	0.29	0.22	27242589247
CyberShot DSC-S75	Sony	$494.99	3.3	3	0.37	0.29	0.52	27242589278
CyberShot DSC-P2	Sony	$390.99	2	3	0.28	0.29	0.2	27242607354

The combination of the exact and dim characteristics within the limits of the general model of data allows expanding essentially opportunities of the interface between the user and information systems. The relational databases traditionally used for storage and search of data about various products, can be used also for storage of the dim characteristics. If to consider, that exact characteristics can be set by group of key values $\mathbf{K_1, K_2 ... K_i ... K_n}$, and the dim values can be set by group нормированных values $\mathbf{F_1, F_2 ... F_i ... F_n}$ inquiries for a finding of demanded record can be incorporated together and presented to a database in a general view as inquiry:

*SELECT * FROM table*
WHERE KEY_i *IS LIKE %*K_i*% ... AND ...*
($FUZZY_i < F_i + delta_i$ *AND* $FUZZY_i > F_i - delta_i$*) AND ...*

where

$\mathbf{KEY_i}$ - the name of a field in the table of the base, containing key characteristics;

$\mathbf{K_i}$ - value of the characteristic (exact);

$\mathbf{FUZZY_i}$ - the name of a field in the table of the base, containing key characteristics;

$\mathbf{F_i}$ - value of the characteristic (washed away);

$\mathbf{delta_i}$ - a range of area in the dim characteristics.

Such kind the universal inquiry satisfies to the broad audience of potential questions which examples have been resulted above. The range in the field of the dim characteristics **(delta)** sets borders of provisional area of values which can satisfy a concrete question of the user. This size gets out proceeding from practical reasons and our examples is equal 0,1, that makes 10 % from a positive range of the dim values (see Fig. 61).

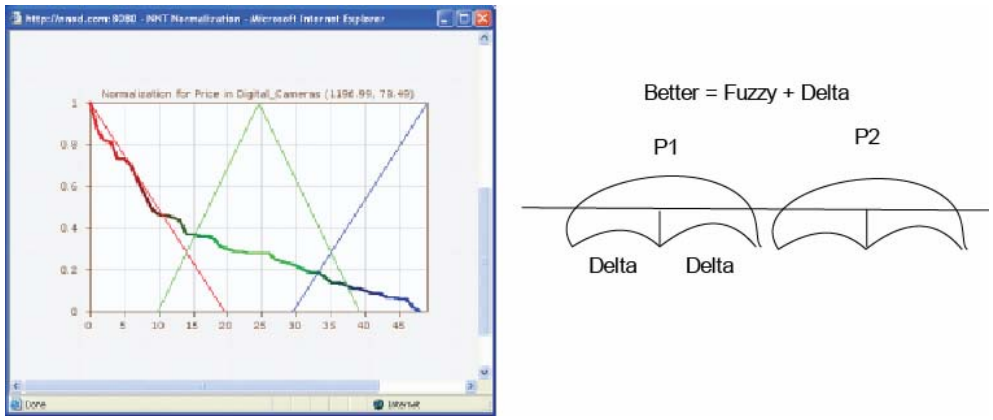


Fig. 61 The fuzzy inquiries to a database

Thus, during each moment of time there is a set of concrete characteristic values which set representation of system that *the buyer* means and on the basis of which it is possible to choose group of the records from a database, satisfying to these representations. We shall name such group *a selective context*. During interaction between the person and information system *the*

selective context will change, but we shall consider what always probably to choose from a database group of records (which can be and empty), corresponding a current *selective context.*

Let's enter for everyone key $K_1, K_2, \ldots K_i, \ldots K_n$, and washed away $F_1, F_2, \ldots F_i, \ldots F_n$ values in neural model corresponding neuron N_j, which excitation, defines: whether the given characteristic value in universal *SQL* inquiry will be included. Moreover, for the dim characteristics the level of excitation of neuron can correspond to size F_i of the dim value, and in this case:

$$F_i = E(N_j),$$

where

N_j - the neuron corresponding washed away нормированному value F_i,

$E(N_j)$ - a level of excitation of the neuron corresponding the given dim characteristic.

For maintenance *of a selective context* it is possible to use neural triggers which allow storing the active dynamic conditions reflecting a current condition of process of interaction. With their help it is possible to create both traditional static binary remembering elements, and dynamic elements, which excitation can change in time according to the internal mechanism of the category.

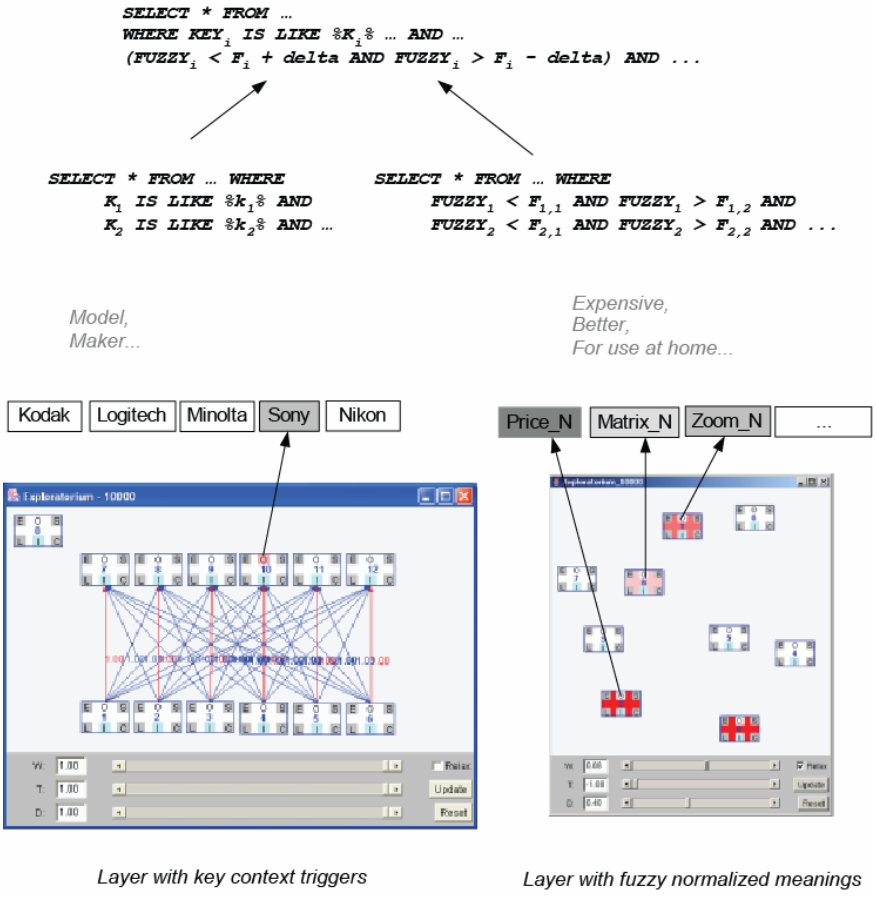


Fig. 62 The neural layers supporting exact and fuzzy inquiries to a database

On Fig. 62 the example of structure of the neural layers, containing two group - contextual triggers for maintenance of current values of one table of a database - exact characteristics (in the given example - *Model/Make*) and the individual neurons storing current values for dim characteristics (*Price-N, Zoom-N*, etc.) is shown.

FEEDBACK IN INTERACTIVE PROCESSES

Finally, efficiency of interactive system is defined информативностью messages which this system is capable to generate during dialogue. As two systems that dialogue between them made sense participate in dialogue at least, each system should estimate quality of created messages and on the basis of it to carry out corresponding updating the behavior. The nature of neural models enables us to make an assumption, that even and as a first approximation, but two systems (the person - a computer), participating in dialogue, perceive messages on the basis of identical principles. Such assumption allows developing the mechanism, capable to receive in a computer the approached estimation, the information processes connected with given conversation occuring inside of the person.

As already it was marked above, *the information is a measure describing changes which occur inside of system as a result of reception of the message*. Such*, at first sight,* very much the general definition, leaves to us a greater freedom in a choice of methods of a finding of these changes. For example, for calculation of quantity of the information it is possible to use the traditional ways based on definition of the information as a measure of change of uncertainty of system. Neurons represent computing elements which ideally approach for registration and storage of such changes. The result of external influence can be in that case calculated as a difference between возбуждениями neurons in the certain area of model before this influence.

Let's assume, that system $\mathbf{S_1}$ initiates process of interaction with system $\mathbf{S_2}$ and thus, in system $\mathbf{S_1}$ there is a function which allows some way to define border of expected result. Value of this function can be presented as change of some size having information sense, and is expressed in bats. We shall consider that the expected result is equal *1 bit* if the prospective answer has two possible equal probable values. In many cases such answer can be shown to one of values of pair *yes/is not present*. If system $\mathbf{S_1}$ manages to formulate a question in such form, that system $\mathbf{S_2}$ is capable to answer it unequivocally from a set consisting of pair of prospective answers actually at reception of such answer it is possible to consider process of interaction completed. System $\mathbf{S_1}$ receives expected quantity of the information, and the purpose of interaction thus is reached, the task of a unique question. However in a real life such situations meet seldom enough. In most cases, the expected result is not reduced to the simple binary answer, and on the other hand, system $\mathbf{S_1}$ not always manages to formulate a question in such form, that system $\mathbf{S_2}$ can unequivocally answer it.

During alive dialogue people constantly specify and correct the purposes and accordingly the form, and the maintenance of conversation. Such behavior is quite coordinated with behavior of cybernetic systems with feedback. Feedback in this or that form are inherent practically in all complex technical, biological and social systems [30]. With reference to economy and the financial markets feedback have exclusively information character, and thus they influence decision-making processes which in many respects are similar to the processes occurring in interactive information systems at a choice of proper responses. Addition of the mechanism of feedback in systems of interactive interaction opens opportunities for application of well developed cybernetic and mathematical methods of the analysis at research and optimization of their behavior. However with reference to neural models it is necessary to consider, that their complexity considerably surpasses complexity of systems which are traditionally considered as objects of management.

Feedback in dialogues directly influences its context. Integration of contextual fields which can be constructed, using neural models, together with the mechanism of the feedback, allowing to correct development of dialogue, qualitatively changes process of interaction of the person and information systems.

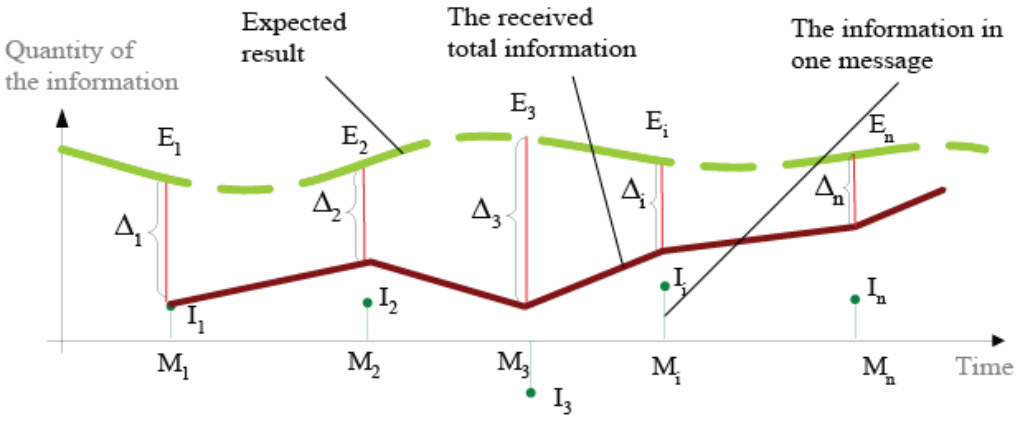


Fig. 63 Change of an information condition of system during an exchange of messages

On Fig. 63 the characteristic of process of dialogue in system $\mathbf{S_1}$ is shown. The expected result can be presented as some criterion function which sets prospective value of the received information. This value for some cases can be known with enough high accuracy. For example, if system $\mathbf{S_1}$ the fact of presence of any product or presence of the certain property at object in this case the expected quantity of the information will be equal *E = 1 bits* interests. The question which system $\mathbf{S_1}$ can set to system $\mathbf{S_2}$, in this case looks as follows:

***System S_1**: At chamber Nikon Coolpix 5700 increase is equal 8x?*

***System S_2**: Yes.*

Such type questions, in which the prospective answer is reduced to a choice from the minimal set of possible variants, refer to closed. Questions, in which the prospective answer can have greater variability, refer to opened.

It is obvious, that if the question will be formulated in a little bit other kind, the expected quantity of the information will change:

***System S_1**: What increase at chamber Nikon Coolpix 5700?*

***System S_2**: 8x.*

In this case, expected quantity of the information it is possible to estimate approximately in *4 bit* (if to assume, that there are 16 variants of increase using the elementary formula of calculation of quantity of the information:

$$\mathbf{E = \log_2(P_{\text{before the message}} / P_{\text{after the message}})}$$

where $\mathbf{P_{\text{before the message}} = 1}$ **and** $\mathbf{P_{\text{after the message}} = 1/16}$

E = 4 *bits*.

If in the answer system $\mathbf{S_2}$ besides the simple answer will add still data on the price, presence of this chamber in a warehouse, and т. Item in this case the received information will be much more than expected *4 bits*. For example, system $\mathbf{S_2}$ can answer the same question as follows:

***System S_1**: What increase at chamber Nikon Coolpix 5700?*

***System S_2**: 8x, and its cost at us $1,047.99. But I shall recommend to you DiMAGE 7Hi Minolta for $1,196.99.*

In this case, we can consider, that this information can appear useful to the buyer, however quantity of the received information, it is obvious, much more, than quantity of the expected information. The answer can contain additional data that from the point of view of system $\boldsymbol{S_1}$ can be considered as information noise. For example:

***System S_2**: It is very good chamber, and this sanction 8x, allows doing wonderful pictures. At us now sale and its cost of all $1,047.99. But I shall recommend you other chamber. I have a friend and it without mind from it - DiMAGE 7Hi Minolta all for $1,196.99.*

From its part system $\boldsymbol{S_1}$ also can bring information dust in a question:

***System S_1**: you know, I heard, that at all chambers Coolpix the good sanction. And best of them is 5700. You will not prompt me, what sanction at it?*

Certainly, interaction of the person with interactive information systems imposes the certain frameworks on the form of dialogue; however the general principles of information estimations of processes of an exchange of the information remain constant. Messages which systems exchange, will contain a surplus information, and they will contain also information noise. It is obvious also, that in many cases we will not manage to receive exact information characteristics of these messages, and however we should aspire to minimization of surplus information at generation of answers. Thus it is necessary to consider, that system $\boldsymbol{S_2}$ can have the representation about the expected information in system $\boldsymbol{S_1}$, and it allows it to build various trajectories of development of dialogue and depending on it to give for system $\boldsymbol{S_1}$ the additional information.

In an initial condition system $\boldsymbol{S_1}$ possesses some information $\boldsymbol{I_0}$ on its interesting event. After reception of message $\boldsymbol{M_1}$ system $\boldsymbol{S_1}$ gets some information $\boldsymbol{I_1}$, and passes in a new condition in which it can send system $\boldsymbol{S_2}$ a new question in reply to which it can receive new additional information $\boldsymbol{I_2}$. Generally received information can lead to that the general representation of system $\boldsymbol{S_1}$ about interesting event can decrease. During each moment of time there is a distance $\boldsymbol{\Delta_i}$ between current information representation of system $\boldsymbol{S_1}$ about its interesting event and the criterion function setting restriction on quantity of the received information.

Thus, the principle of a feedback with reference to systems of interactive interaction can be formulated as aspiration to minimize distance $\boldsymbol{\Delta_i}$ in each point of process of dialogue, as **by** system $\boldsymbol{S_1}$ initiating questions, and system $\boldsymbol{S_2}$ answering these questions and simultaneously capable to ask additional questions from its part. Symmetry of systems $\boldsymbol{S_1}$ and $\boldsymbol{S_2}$ allows considering these systems as equal in rights. Really, system **S1** something can interest in system $\boldsymbol{S_2}$, and in turn system $\boldsymbol{S_2}$ can have interest to the information which system $\boldsymbol{S_1}$ possesses.

As system S_1 and system **S2** can have the own various purposes, and thus should model and supervise both own behavior, and behavior of the partner on dialogue, the problem of achievement of effective interactive interaction is reduced to search of an optimum trajectory of dialogue in conditions of existence of two criterion functions setting optimum information corridors for systems S_1 and S_2.

CONSCIOUSNESS AND SUBCONSCIOUSNESS

Sequences of messages which two systems can exchange form a trajectory which can be presented as sequence of transitions in some space of all possible conditions **P** in which can there is a given system. In each present situation of time condition $\mathbf{P_i}$ in which there is a given system, we shall name an active or conscious condition. For each possible condition $\mathbf{P_i}$ it is possible to create a corresponding card (page) $\mathbf{C_i}$ which can define reaction of system. Card $\mathbf{C_i}$ is the container of data and programs and can be loaded into the interpreter of a current condition (*an active field*).

Analogue of space **P** is the Internet consisting of set of pages which become active only during that moment when they are loaded into the browser *(Web-browser)*. For users the Internet, during each moment of time in a current window *the web-browser* active is one page from which hypertext transitions in different directions are possible. Using such analogy, we shall consider, that on each step of an interactive exchange, inside of each system there is an object similar on the properties to current page a web-browser on the Internet and possessing ability to generate the answer and to carry out dynamic transition to the subsequent object. Neural models thus serve as the mechanism providing, on the one hand, preservation of a context and the semantic maintenance of process of an exchange by messages, the answer necessary for generation, and with another, allow carrying out transition to the following object reflecting a new condition of system.

Cards of conditions generally can represent complex structural associations of programs and data, хранимые in format *DHTML/XML*. For editing and viewing of contents of cards the Web-browser by means of which the author can carry out all the basic operations of editing and updating of their contents can be used. Various sets of cards are resulted as an example in section «Demonstration of work of the linguistic neural environment» . Received as a result of *the XML-description* can be kept in any suitable system of a data storage to which can have access the interpreter of a current condition. Unlike a web-browser, the interpreter of current conditions carries out actions, basically connected with change of inwardness of all system. It cooperates with neural model, defines conditions of transition in a new condition and a choice of a following card, loads programs for dynamic execution, accepts entrance messages and transfers target messages in corresponding channels of input/conclusion. As a whole, its functions are limited enough, that allows to consider it as a “rigid” or built in component of system. On the other hand, the formats of cards containing in them of the program and data, can be somehow complex, that does system opened for updating and expansion.

The interpreter of a current condition (64) can be considered Fig. as analogue of human consciousness. During each moment of time in an active floor of the interpreter the current received message contains and the answer to this message prepares. In this field entrance and target neural layers converge, and it has access to all set of neurons within the limits of active neural model. In turn it is possible to consider neural models analogue of sub consciousness. In them there are massive parallel processes which cannot be reduced to representations which can be expressed in the form of analytical expressions in the habitual form.

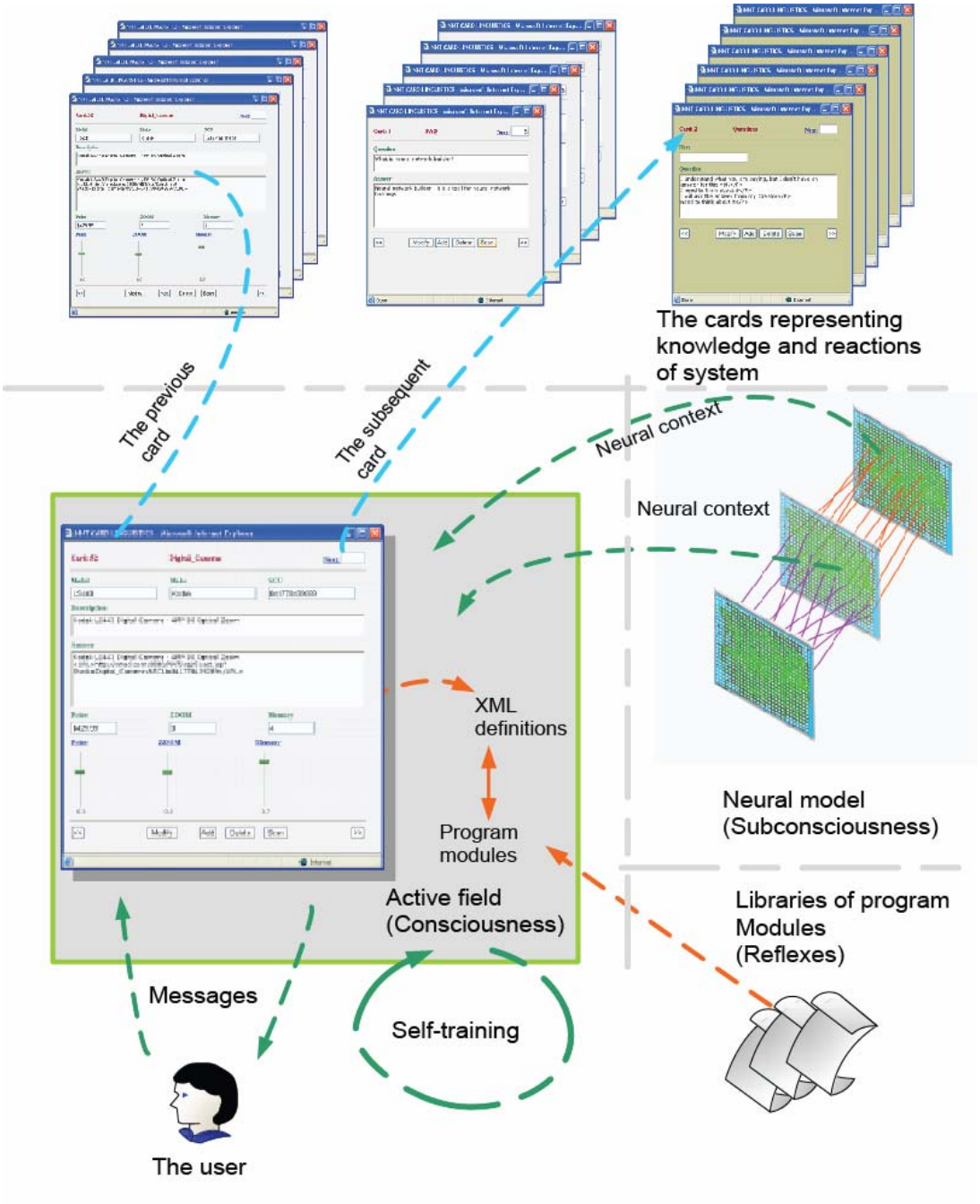


Fig. 64 Communications between objects in the Neural Environment

Connection of neural models with an active field, in which interpretations of various programs are possible, opens opportunities for integration of algorithmic and neural methods into a uniform complex. Performance of the programs, interpreted or preliminary откомпилированных, and dynamically loaded modules (*Java-classes* or *DLL-objects*) it is possible to compare (scripts) to reflex knowledge. Cards and *XML* carry out a role of the special connecting mechanism in integration of algorithmic and neural methods. *XML* is the universal and open language for which

it is possible to create rather easily various interpreting systems, and cards possess the intuitive user interface allowing external users to support and develop integrated system.

Algorithmic methods allow to connect to neural models a wide set of solver-agents for the analysis of speech, the decision-making, traditional numerical and program problems and т. Item Integration of these two various ways of the modeling based on essentially various axiomatic, is possible only by their artificial association, and for these purposes and the active field serves.

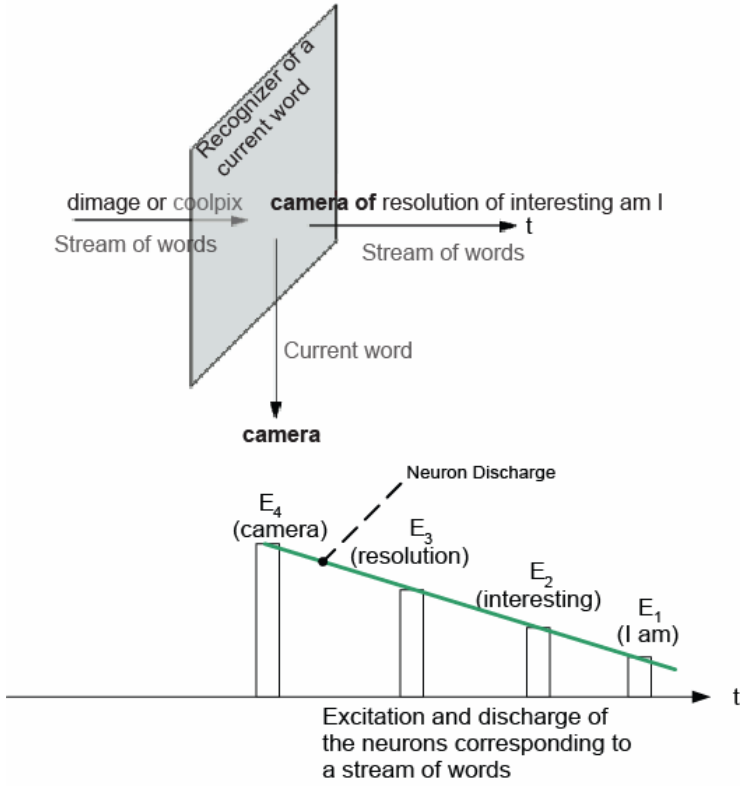


Fig. 65 Sequence of words in a stream of speech and corresponding excitations of neurons

Such integration allows solving effectively, for example, the problems connected with understanding of various forms of speech. In perception of oral and written speech is much in common, but during too time they essentially differ on dynamic characteristics, on time of reaction and on grammatical properties. During conversation participants communicate, corresponding the certain theme. In that case when one of participants asks a question, from the second participant reception of the answer is supposed, and thus time during which this answer should be is received, limited rigidly enough. Reaction to the received messages occurs in real time, and the stream of the words defining given conversation, as a rule, cannot be repeated all over again.

On character of interaction and time of reaction, the exchange of messages can be reduced to three basic forms. To conversation and the contextual layers, allowing to support active those words or values which concern a present situation. During conversation of a word of an active layer receive primary excitation which in due course dies away if from contextual layers this word will not receive the additional excitation confirming importance of this word by the current moment.

Alive speech. The message is not complete and not accessible in a text form. There are no the precise borders dividing phrases. Time of response is critical. The context depends on time.

The Chat. Time of reaction is limited; the message is accessible to the analysis. The context of the concrete message can depend on the previous messages. Phrases are divided enough, though there are infringements of the standard grammatical rules.

Text messages, e-mail. The message represents the finished sequence of offers. Time of reaction is not critical.

Chats or exchange of messages in a text kind borrow intermediate position between dynamical oral conversation and static process of reading of texts. In chats as well as in conditions of oral speech, there is limited time of reaction for messages; however messages in a chat represent, as a rule, quite finished grammatical designs.

In oral speech the stream of words does not form the finished phrases having precisely expressed grammatical forms. Offers can be allocated in this stream by rather conditional image, unlike chats or usual static texts. Division of grammatical designs in oral speech is much more complex, than in written. Besides such factors as the intonation, special a turn of speech and pauses, time are the important attribute, allowing to define borders of offers and to establish a context connecting separate, sometimes enough отстоящие from each other, words on sense. On Fig. 66 the structure of communications between an active layer in which the current words concerning to concrete are stored is shown

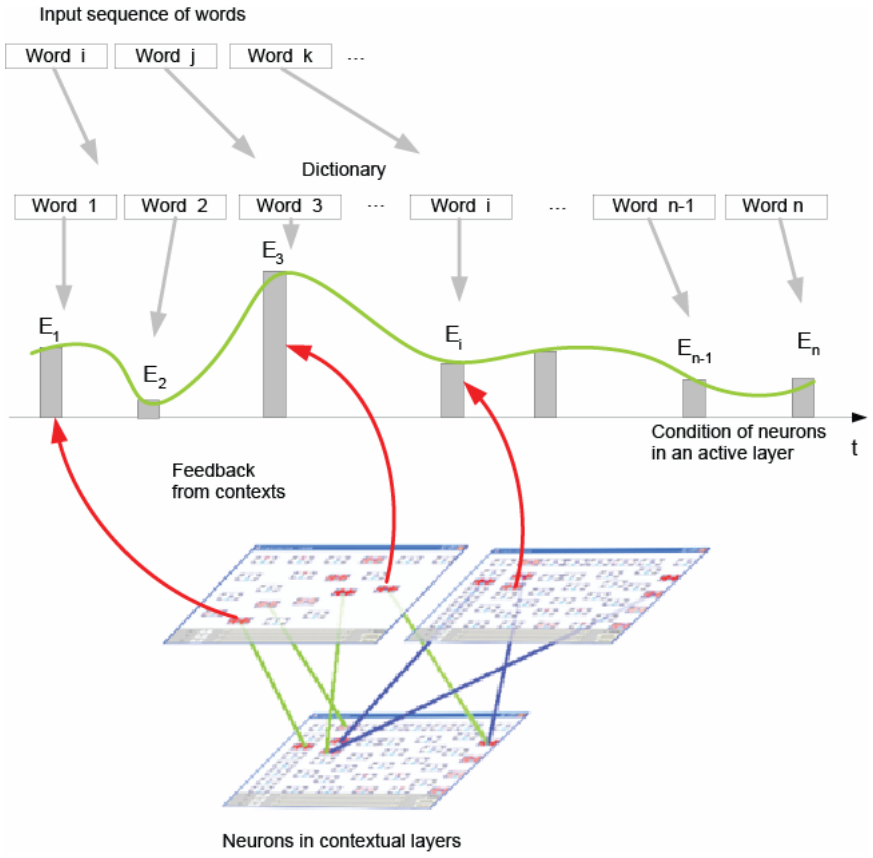


Fig. 66 Stream of words and feedback

The neural systems perfectly adapted for display of a plenty of dynamic processes, ideally approach for modeling dynamics of change of sense of oral speech. Thus the context of conversation, representing a feedback mechanism, can influence as words or values, хранимые in an active layer, and on the connected values inside of contextual layers.

Character of interaction influences structure of speech of the person without dependence from that, is its interlocutor other person or a computer. In one of early works on this theme [6] changes in structure of speech of the person depending on a way of the communications are considered. Authors consider conversations between two people in direct dialogue, by phone and through the computer terminal (analogue of a modern chat). Analyzing situational conversations between people, in clause the interesting conclusion in occasion of use of direct contextual turns - words *THIS*, *IT*, *THAT* and т is resulted. The item In a chat such contextual indexes are used much more often, than in telephone conversations. It is obvious, that the opportunity to see the previous messages allows the person to define quickly sense of contextual communications and by that to reduce time for transfer of repeated words or word-combinations, due to use of short contextual indexes.

During oral alive conversation, the person does not have opportunity to return to the previous messages, however the computer is capable to store all sequence of the acted words and to support dialogue in habitual for the person to the form, computer systems should possess ability *"to forget"* the previous messages to similarly how it is done by the person. Such ability to «забыванию» in neural models is modeled due to the category of neuron. The example showing excitation and the category of neurons in an intermediate layer as a result of influence of sequence of words is shown on Fig. 38.

In that case when as the partner of the person in oral dialogue the computer acts, preliminary recognition is carried out, as a rule, by acoustic system of recognition of a voice (*Automatic Voice Recognition - AVR*) which then transfers sequences of words in linguistic system of recognition of speech (*Automatic Speech Recognition - ASR)*.

The majority *of systems AVR* uses the grammar, allowing to distinguish words in a context of the certain offer. Application грамматик considerably raises quality of recognition due to addition of the likelihood characteristic of a word in sequence. The problem of recognition of speech consists in search of such combination of acoustic models together with linguistic at which the most effective definition of probabilities of word-combinations in sequences of acoustic forms of words and dictionaries corresponding them is provided. When grammar is well set, modern acoustic распознаватели allow to receive exclusively high accuracy of recognition.

It is obvious, that to expansion of a subject domain of conversation there is an increase in complexity грамматик, and the problem of modern systems of recognition of speech consists in it. It is practically impossible to create the universal grammar, capable to provide recognition of sequences of words for all possible variants of human messages, even for rather narrow subject domains. Among various ways which can be used at the decision of this problem, it is possible to allocate statistical models (*Statistical Learning Models - SLM*), capable самообучаться during preliminary training (see, for example, [5]), and methods of the adaptive management, allowing to allocate natural sequences and to be trained by means of well balanced algorithms and criteria [33].

One of ways of generation acoustic грамматик lays in application of the neural models, capable to establish interrelation in sequences of words (or their elements representing). On Fig. 67 the

result of training of a dictionary layer of neural model on sequence of the offers taken from reports of chats, such, for example, as is shown:

What's Longhorn?
Who are you?
Tel me about your company
Tell me about your technology
How long have you been in business?
What's your phone number?
What's his phone number?
...

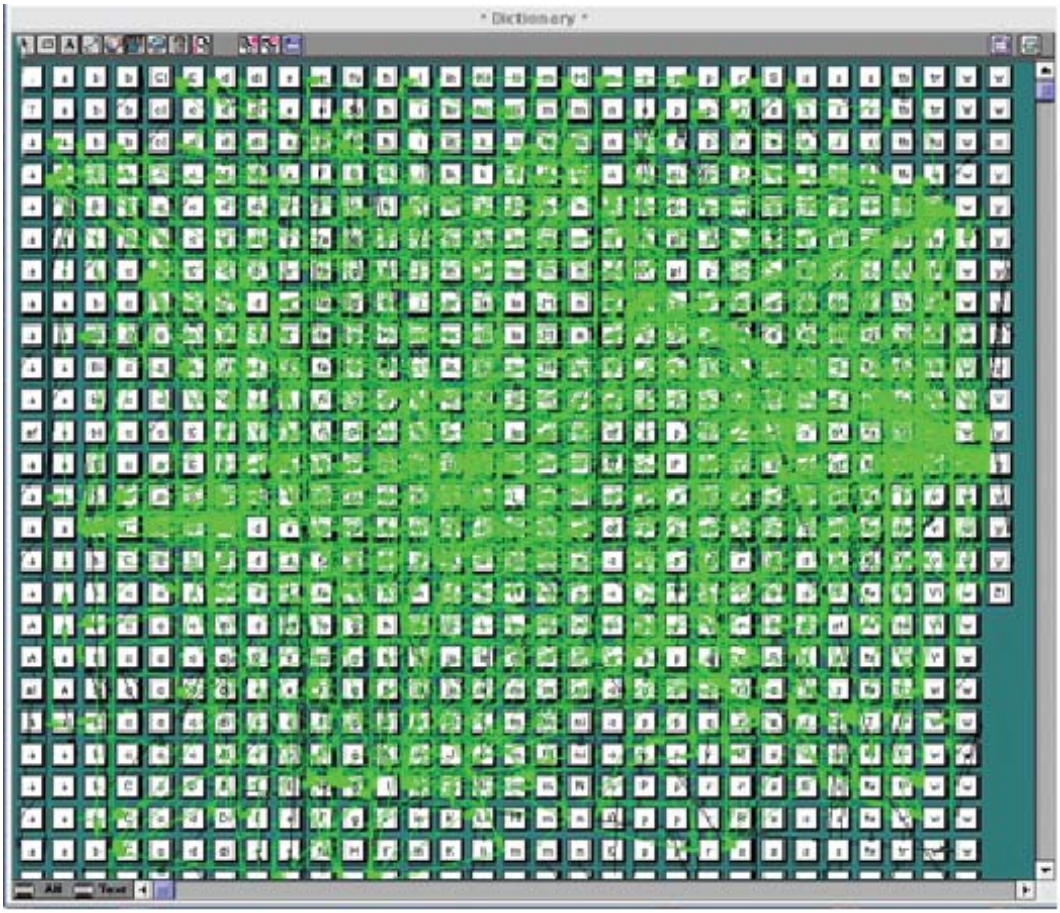


Fig. 67 Statistical dependences between words after training (for presentation in words initial letters are shown only)

Training of neural model consists in construction of the initial dictionary, and then in an establishment of communications between words of offers and the subsequent updating of these communications in process of training on the subsequent phrases. There is a set of algorithms of training of neural networks which can be easily applied to our model.

TYPICAL CONSTRUCTIVE ELEMENTS

The wide circulation and, finally, success the Web are caused by a successful combination of technological opportunities of two groups directly participating in creation of sites - the experts who are responsible for the maintenance of pages, and programmers who do these pages functional and active. The range of editing tools of contents the Web of pages allows the experts who are not having skills in programming, to publish knowledge in that form which is close to traditional methods of publications of materials in a paper kind. Terminology and editing tools are based on such accessible, practically for all formed people, concept as *page*. On the other hand, program components which can be built in page, represent a wide spectrum of opportunities - from the elementary scripts written on *Visual Basic*, up to rather complex systems of dynamic generation by means of *CGI* modules that allows to coct the Web of a site of programmers of the most various qualification and specialization to process of designing.

Such features of the organization and technological receptions a web-design, can be quite applied in neural modeling. The certain part of this process is connected with representation and management of subject knowledge, and experts from various areas, as a rule, not being are engaged in professional programmers or mathematicians it, first of all. These experts form dictionaries, define sense and interrelations between contents, prepare answers and т. The item For them, neural models are presented in the form of various conceptual editors. One of kinds of conceptual editing is use *of cards* which it is possible to consider as analogue of pages a web-site in the Internet. Cards can form a basis for accumulation of knowledge in the form of, habitual for experts, in the same measure, as well as the image of pages is habitual at creation of text and graphic materials in an electronic kind.

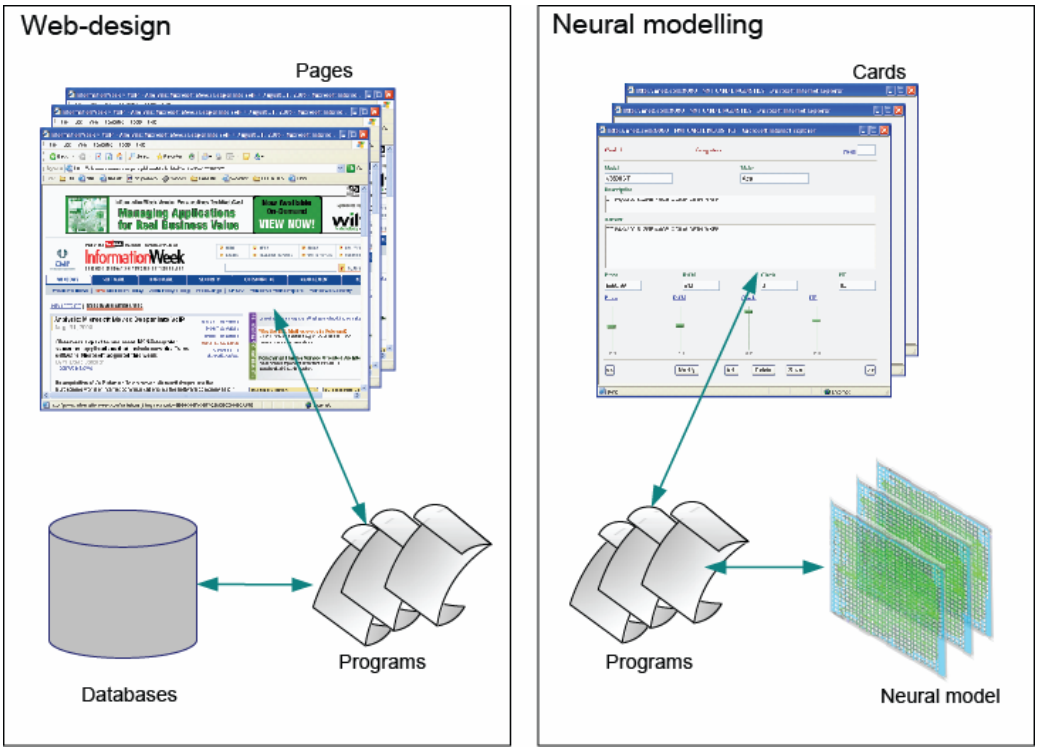


Fig. 68 The Web-design and the neural modeling

Other part of neural modeling concerns to area of professional programming, here again, as well as during a web-design, programmers can fill this process by dynamic contents. In programming neural models it is possible to allocate following basic directions:

- Creation of interfaces to databases and integration of results of search into generated answers;
- Programming various methods of recognition and classification, with reference to a concrete neural kernel;
- Creation of background programs for management of dialogue;
- Creation of sequences Neuro-DNA for cloning of necessary neural structures.

Just as in algorithmic programming where there are base algorithms, methods of construction of programs, in neural programming exist similar ways of interpretation and corresponding structures. Practical realization of the interactive systems using neural models in many depends on their integration with traditional algorithmic methods and programs. Thus, besides reports and the interfaces allowing various components of system to exchange by data, the special role belongs to typical elements and structures which use can essentially reduce time of development.

In a stream of messages which interactive systems exchange, always it is possible to allocate groups of the interconnected offers between which there is a certain contextual communication. In alive speech response depends as on the information containing in the received message, and from an accessory of this message to the certain linguistic group. For example, in Russian of the offer share on *narrative*, *exclamatory* and *interrogative*, however it is possible to expand this classification having entered additional groups, such, as *requests*, *questions*, *statements*, *answers*, *statements* and т. The item the Accessory of the received message to this or that group and the context connected with it define the subsequent responses of system. For storage of conditions and the contexts reflecting dynamics of interaction and capable to influence responses, it will be necessary for us various распознаватели and remembering elements among which we shall distinguish *the mutually exclusive* contextual elements *independent* and *complementary*.

Mutually exclusive elements define group of events or conditions for which in neural model, during each moment of time, active value one neuron corresponding to thus event possesses only. For all other neurons of the given group of a condition are established in inactive value. *Mutually exclusive* elements are modeled in the form of *the n-humeral* trigger (see Fig. 69).

As example of mutually exclusive group the type of the offer, model of the chamber, the manufacturer, etc. If the buyer wishes to buy the certain chamber and does the following statement (or answers a question):

The buyer*: Me chambers Sony interest,*

that in one of layers of the neural model, responsible for storage of a context *"Manufacturer"*, the unique neuron corresponding value *"Sony"* should be established.

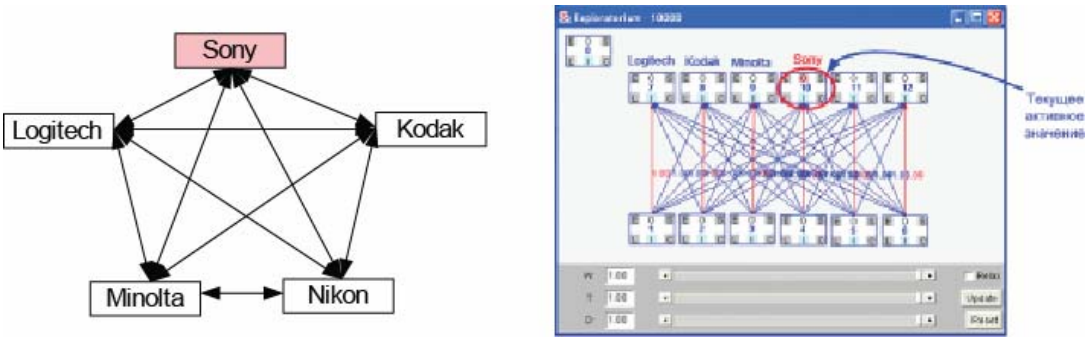


Fig. 69 Structural representation of a mutually exclusive set of values of a context "Manufacturer"

On Fig. 69 structural representation of this context and the neural embodiment corresponding this structure realized by means of group of 6 triggers is shown. It is obvious construct one such contextual layer we can to take advantage of the cloning considered above in the chapter *«Contexts and time logic»*.

Apparently from the resulted examples, it is possible to use the same neural structure for modeling mutually exclusive contextual group for types of offers, the manufacturer of production, model, etc.

By means of the mechanism of cloning Neuro-DNA we can *beforehand "grow up"* (see Fig. 70) enough of contextual layers and to coct them to corresponding structurally functional groups as required, or to add a new layer during that moment when in it there is a need. It is possible to assume, that during development of a human germ there is something similar - the organism grows up necessary structures in the future and during a life will involve as required under corresponding contexts. If to assume, that development of a brain occurs in a similar way, the hypothesis that the human brain is used only on 10 % receives quite objective substantiation - if biological structures are prepared for the future connections in this case redundancy of system should be sufficient that stocks of neural groups was enough during all life.

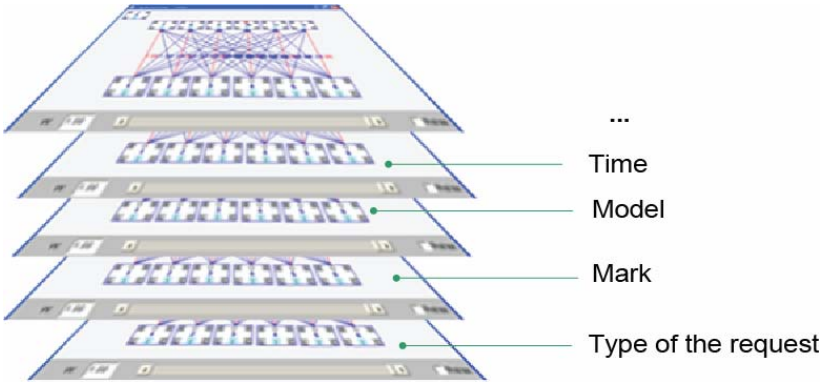


Fig. 70 A stack of mutually exclusive contextual groups

Ready contextual layers can be attached to outputs of distinguishing layers which are responsible for definition of sense of a current phrase. Outputs of contexts can be used as feedback for distinguishing layers or to form the basis for generation of answers, in programs-interfaces to databases, etc.

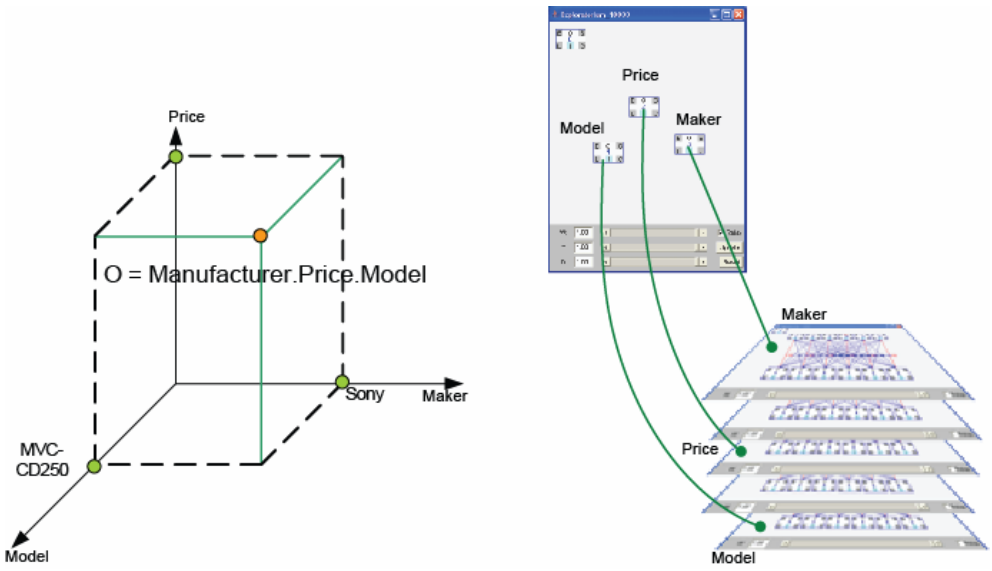


Fig. 71 Complementary contextual values

Complementary values of contextual layers allow creating complex structures on the basis of various characteristics and properties of a subject domain. On Fig. 71 the coordinate space of the complementary values describing representation of the buyer about the chamber in a present situation of time is shown. Current values *of mutually exclusive* contexts at higher level can form groups *of complementary* values. Checking a condition of corresponding contextual layers, the system can always know, that is concrete at present time the interlocutor means. It is possible to consider, that *the Manufacturer*, *the Model* and *the Price* are names of variables which the programmer can use, for example, at construction of references to databases.

Though neural models possess active properties which essentially distinguish them from traditional databases, however practically all operations, применимые to databases, применимы and to neural models. So for example, operator *SELECT* allows to receive values of concrete neurons or groups of the neurons belonging to certain кластеру (layer).

Just as there is no universal way of design a web-pages, there are no universal methods of designing of neural models - neural кластеры it is possible to connect arbitrarily. We shall consider one of examples of construction of the linguistic analyzer which is used in system of construction of virtual assistants. Examples of linguistic models are resulted in section « Demonstration of work of the linguistic neural environment » *NNE*, and on Fig. 72 the block diagram of interaction of various functional neural layers during the analysis of the phrases acting from external system is shown.

At the very first stage of a word of which the phrase consists, consistently raise neurons of a dictionary layer. All neurons, which level of excitation has reached threshold value, are designated in this figure as possessing by a token which can be transferred further, on groups of the layers which are carrying out a role of primary recognition. Распознаватели, depending on combinations of words and contextual values, form intermediate target values which form a basis for redistribution возбуждений in groups of contextual layers. Contextual layers form the groups of neurons which are carrying out the basic functions under the analysis and formation of so-called *"understanding"* of a current situation. On the basis of their condition target reactions are developed and formed excitation in feedback which will define recognition of the subsequent

phrases. Reactions, finally, form excitation in a target layer with which programs of generation of answers work. Neurons of a target layer, on the one hand, define what particularly the program will be caused for processing and generation of the current answer, and with another, prepare for this program the certain linguistic values, such for example, as the price, qualitative characteristics and т. The item of the Program also can receive current values of contextual layers and to use them for generation of inquiries in databases.

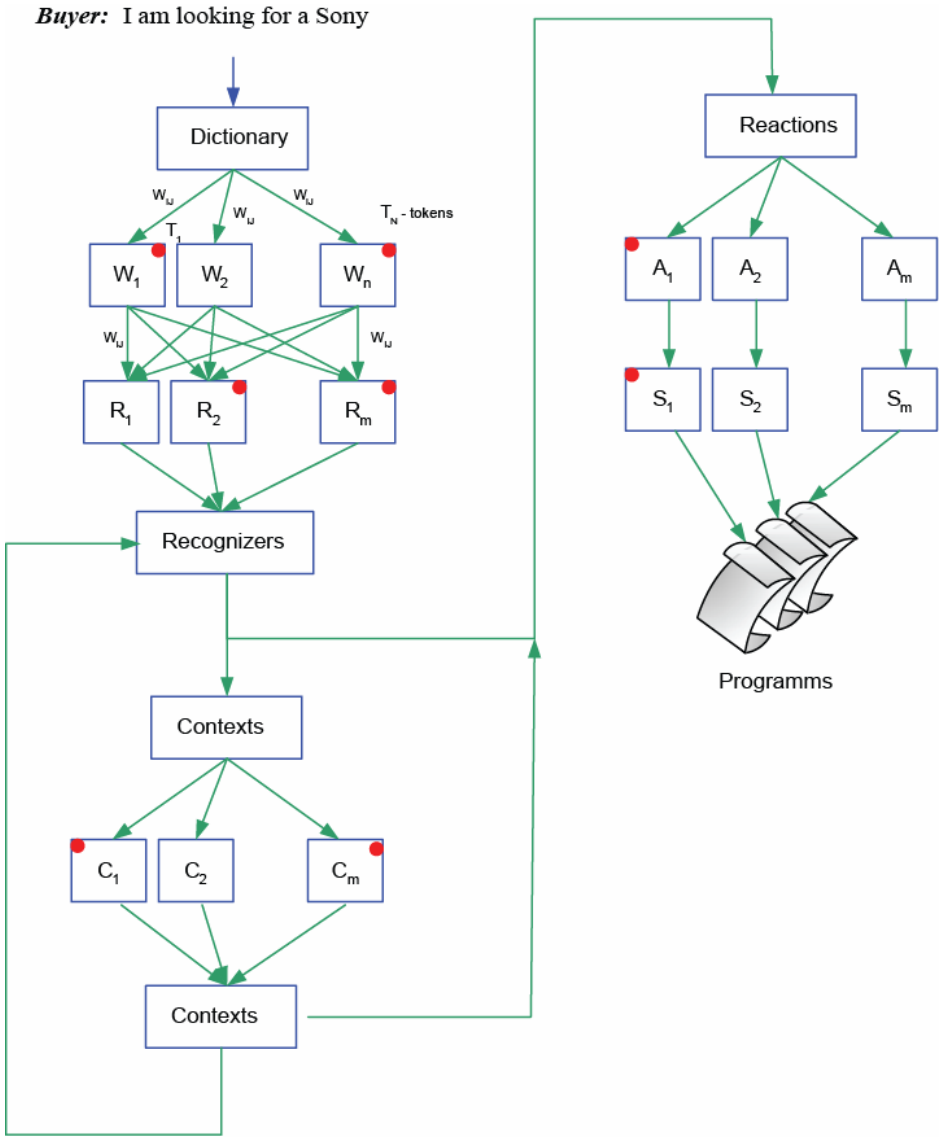


Fig. 72 Interaction of neural layers during the analysis of offers

The important property of interactive systems consists in necessity to follow logic sequence of questions and the answers initiated both one, and other party. Neural models possess the effective mechanism, allowing supporting logic of dialogue absolutely distinct from classical the block-circuit of the approach in the way.

Let's understand as *"question"* such message $\mathbf{mQ_i}$ which system $\mathbf{S_1}$ sends to system $\mathbf{S_2}$, that after it посылки system $\mathbf{S_1}$ ingenerates itself process of expectation of system $\mathbf{S_2}$ of the reciprocal

message $\mathbf{mA_i}$. Differently, after посылки messages $\mathbf{mQ_i}$ system $\mathbf{S_1}$ ingenerates itself the process, containing operator **WAIT (e_i).** Generally process **P (mQ_i)** and operator **WAIT (e_i)** can depend on time, and after certain time process **P (mQ_i)** can stop a condition of expectation.

One of specific features of interactive processes consists that each of the parties taking part during dialogue can initiate messages independently and not necessarily in a context of the previous exchange. The essence of an asynchronous exchange of messages consists that the question and the answer at an exchange between people are not strictly connected events. Below the characteristic fragment of conversation in which the question of system remains without the answer is resulted and the reciprocal offer consists of two parts which should be interpreted depending on a context:

System: *Hello, how are you?*
The person: *Greetings, me interests weather for tomorrow.*

Or in case dialogue is initiated by the person:

The person: *Greetings, me interests weather for tomorrow.*
System: *Hello, tomorrow it is expected.... Yes, by the way, would you like.....*

The decision by means of neural model allows using the approach based on an opportunity of a choice of the subsequent question on the basis of the most active condition of communication between current neuron and all possible continuations. On Fig. 73 the scheme of a choice of a following most suitable question from a current condition is shown.

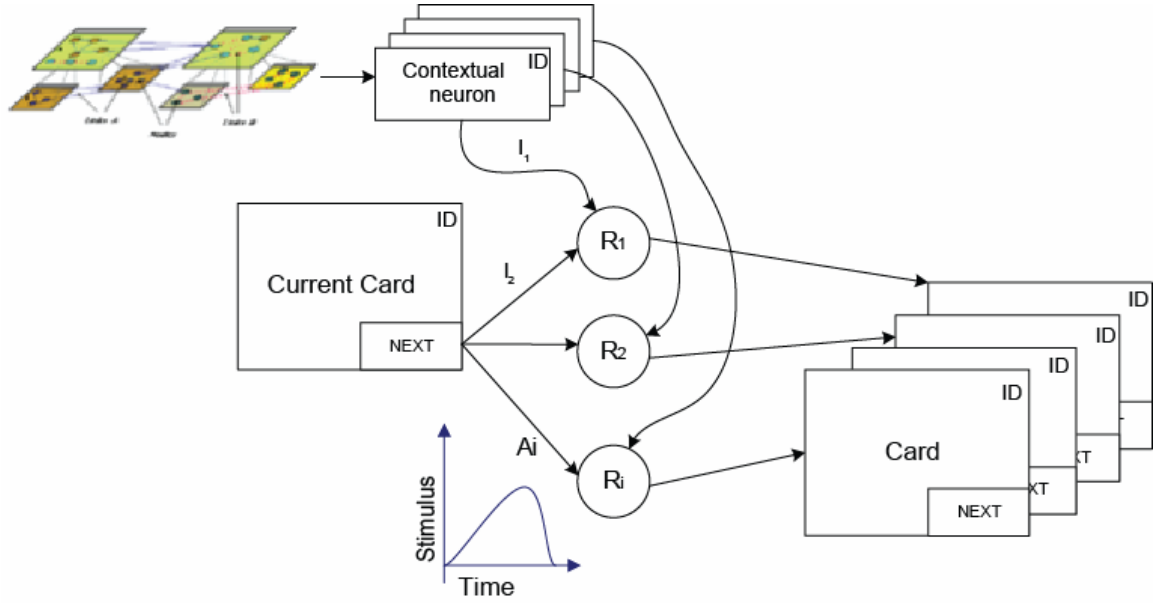


Fig. 73 Generation of sequence of questions and answers

Proper response of system can essentially become complicated if to consider, that time necessary for reception of the answer, both from the person, and from information system, can make a long interval, quite sufficient that for this time it was possible to formulate and enter the next message. Consecutive programming of such dialogues differs unjustified complication of logic even for cases with insignificant quantity of prospective reactions. The problem of processing of the

distributed expectations *(Wait)* can be solved by distribution corresponding excitations in neurons of model. Fairly it is considered, that the most complex in neural modeling is creation of multilayered distinguishing and reacting structures. However by means of cloning with application *Neuro-DNA* this process can be essentially simplified, as the structures created once can be made multiple copies, transferring reproducing sequences from one model in others.

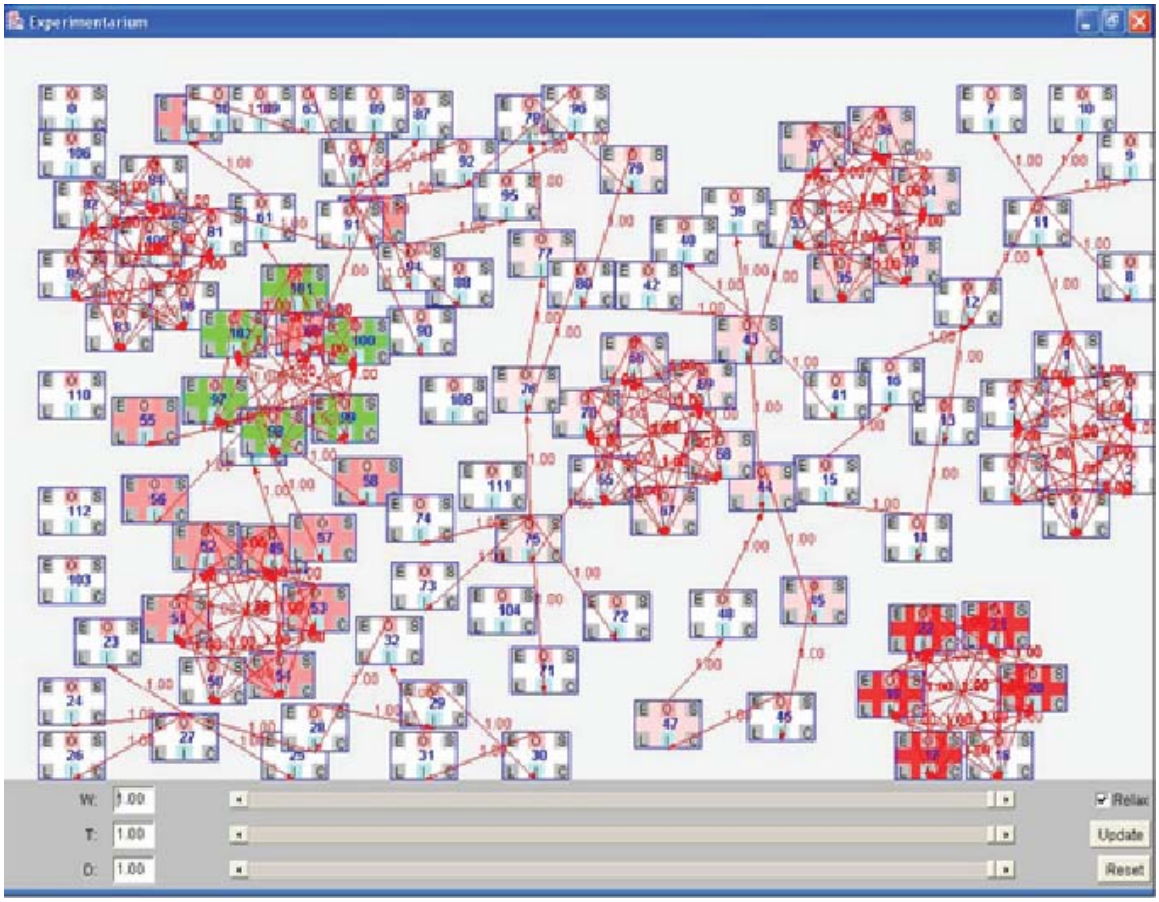


Fig. 74 The model of the basic typical dialogue structures which have been "grown up" by means of the mechanism of duplication on the basis of Neuro-DNA

On Fig. 74 the Example 5 in which the same sequence *Neuro-DNA* is used at cloning by group of neurons is shown, that does possible creation of practically unlimited models on complexity. It is possible to assume, that addition of the mechanism of mutations in neural models will lead to occurrence absolutely unpredictable on the properties and behavior of systems.

THE CONCLUSION

It would be much easier to speak about neural programming if its bases could be reduced to the habitual standards accepted at definition of traditional programming languages, such as *Java* or *C++*. Quite probably, those definitions will be already received in the near future. It is obvious, that it is possible to construct set of theoretically equivalent programming systems in which there will be ways to solve other by all the problems considered in this book. Probably, someone can seem strange and unusual a way of programming which consists in connection of elements with each other or in selection of the certain combinations of time intervals and weight factors therefore there is a unique structure, in the unique way of which check of working capacity, experiment is. However it is possible to hope, that will be also such who will see in such way of programming an opportunity to create the systems possessing surprising internal harmony and at it bringing real practical advantage.

Neural programming is a way of construction of neural models, and at the same time a way of management of interpreting mechanisms by means of which these models can be executed. If to follow traditions in programming languages, it would be necessary for us to construct first of all grammar of neural language and then to offer structure of the interpreter, capable to execute neural programs. However, on expression Neumann's background - *«Language of a brain, it not language of mathematics»* [19]. Elements of which neural models consist, essentially differ from what traditional computing systems are collected and as methods of their formalization essentially should differ. Neural models possess one conclusive advantage - they are unusually simple and intelligible and in it between them and music is surprising similarity. Unlike programming languages which development usually assumes necessity of studying rather great volume of preliminary theoretical knowledge, elements of neural modeling just as music, can be mastered very quickly practically by each interested person. As well as in music, in neural models simple results can be received practically at once, and in process of complication of a musical composition or neural model, quantity of new knowledge and quantity of efforts which need to be spent thus, increase proportionally. Linear dependence between complexity of model and necessary efforts will well be coordinated with a principle known in philosophy of a science as Ockham's razor.

Ockham's Principle can be formulated as follows: *«it is not necessary to enter essence over necessity»*, and if to start with this principle it is difficult to find more suitable on the simplicity and at the same time unusually constructive functional element, than neuron. In spite of the fact that neural models are based on very small set of axiomatic concepts, on their basis rather complex and productive systems can be received. If to continue comparison of music and neural modeling and to take for a starting point complexity and harmonious integrity of symphonic products it is possible to see as borders of an opportunity of individual neural programming can far reach.

In neural models, using the same constructive elements, it is possible to construct as the elementary answering machine, capable to distinguish sense of the messages received on a home telephone number, and complex self-trained system of search of the helpful information on the Internet. Expansibility and интегрируемость such models is based on simplicity of interfaces and reports of interaction of neurons. Integration of neural models into the Internet is represented the most interesting problem on prospect. Just as the Internet acting in a role of the huge accumulator of knowledge, saved up by all mankind, neural models are capable to reflect and keep individual knowledge of the concrete person. The micro model of individual knowledge,

being connected to macro model of knowledge, on the Internet forms the system, capable to strengthen work of a brain in the same degree as modern power machines are capable to strengthen work of muscular forces.

In the beginning of All It which эльфы named Illuvator, has generated from ideas Aynurs, and following its plans, they have executed Great Music. And in this Music the World has arisen...

Dj. R. R. Tolkien "Silmarillion"

THE APPENDIX

The neural Experimental Environment

The readers having access to the Internet, can establish the neural emulator on the personal computer and take advantage of additional demonstration materials which are located on a site of the author:

http://www.nnod.com/np/
To get access to the information placed on this site, it is necessary to enter a name of the user and the password. Registration serves only for gathering the statistical information on visitors of this site.

Name of the user: book
The password: guest
The neural emulator or *Neural Experimental (ìØœ Environment* consists of *the Kernel* (*Neural Core*), written on language *Java*, and the user Web-interface representing a set of pages, containing dynamic expansions and functions, написаные in language *JavaScript*. For normal work *NNE* it is necessary, that on a personal computer virtual *machine Java* and the program for viewing a web-pages, for example - *Microsoft Internet Explorer* has been established. Installation NNE is reduced to performance of several steps on installation then it is ready to work.

Step 1: Installation Java

And works with virtual *machine Java* are necessary for installation base knowledge on programming and the general data on this product which can be found on a site *of company Sun Microsystems* - *http://ru.sun.com/java/*
The current version of virtual Java environment can be received to the following address:

Step 2: Reception and installation of the Neural Experimental Environment

Current version *NNE* can be received to the following address:

http: // www.nnod.com/np/

Exploratorium.zip contains a folder *neuron* which after разархивирования is necessary for placing in any suitable place, for example on a desktop of a personal computer. Contents of a folder *neuron* are shown below:

http://java.com/en/download/manual.jsp

After installation *of Java environment* in operational *system MS Windows* to check up its working capacity it is possible as follows:

1. To cause the program the Interpreter of commands - Command Prompt.
2. In a window of the interpreter to enter a command ' java-version '.
3. If Java it is established successfully, there should be a following message:

Step 3: Work with demonstration examples

All the examples showing work neural дра, are organized in the form of folders with оответствующими names. Each folder contains a file *neuron.jar* which is сполнимым archive *of Java-classes* for which start it is necessary to execute a command:

Java-cp neuron.jar neuron param0, param1, param2, param3

Parameters represent values by default internal variables of a kernel:

param0 - weight factor of communications - $w_{i,j}$;
param1 - time of the category of neuron - D (Discharge-Time);
param2 - a threshold of operation of neuron - T (Threshold);
param3 - time of the category of a condition "is allocated" - delay-S.

This command can be given out directly in a window of the interpreter of commands (*Command Prompt*) or is executed as a result of start of a corresponding file *start.bat.*

After start of a kernel if all installations are made correctly, should there will be a window of the interpreter of commands and a window of a neural environment. The window of a neural environment has the name

Exploratorium-10000, where *10000* - number of port, which ассоциирован with the first neural layer presented in this window. In the subsequent examples additional windows-layers can be opened, the corresponding port *10001* will be connected with each of which, *10002*, etc. These ports should be free and resolved for an exchange with appendices inside of the given computer.

In the resulted examples parameters *param0-param3* can accept various values. In the Example 1 *(Example-1)* these values are established as follows:

java -cp neuron.jar neuron 1 3000 -1 10000

In this case:

W = 1;
D = 3000;
T = –1;
delay-S = 10000.

These values can be changed during work of a kernel by means of слайдеров, located in the bottom part of a window of a neural environment then they can be appropriated to any way chosen neuron. As values of parameters by default influence time characteristics of processes and are specific to each example, it is recommended to close *the Java-appendix* and to restart it anew,

using for this purpose a command *start.bat* from a corresponding folder: *Example-1, Example-2, Example-3*, etc.

In each folder besides the interface files *loader (i) .htm* and in addition loaded - *Example-(i) .htm* and *blank.htm* contain. These files contain active components - scripts and show opportunities of interaction of a neural network environment with external appendices. In these pages references to an environment with use http the report are included where as a server the initial name of a local computer *localhost* and internal port *10000* is used:

URL = Host + o-Port.value + "/1=c";

parent.Background.location.href = URL;

That corresponds to the reference:

http://localhost:10000/1=c

In some examples except for the set forth above files, there are files *neuro-DNA*. In an example 4 - *0.dna,* and in the Example 5 - *0-9.dna*. These files are contained with time delays and other data necessary for cloning of linguistic structures, corresponding neurons.

The result of this reference will be placed in *0* frame with name *Background*, dynamically created by page *loader (i) .htm.* After start of a kernel and the interface pages, on the screen of the monitor should there will be a window of a browser, near to a window of a neural kernel then all components of system are ready to work. In case in a personal computer protection *Firewall* is used, it is necessary to resolve work of active components and corresponding ports.

GLOSSARY

Adequate - corresponding communicative installations of the subject during dialogue or perception. This term serves in philosophy for definition of fidelity or differently, qualities of perception. In this sense the true is the absolute of adequacy of thinking to life.

Bits - a unit of measure of the information. Corresponds to change in a condition of system, as a result of reception of the elementary message on event which probability is equal 0.5.

Harmony - harmonious and proportional communication of components in a single whole. In music, harmony is based on laws and rules association of sounds in accords. Unlike a melody, harmony defines the "vertical" coordination of sounds.

Given - the message which has been written down in any form and on any carrier.

The information - a measure of change of a condition of system, as a result of reception of the message from an environment. It is measured in bats. The information is often used as a synonym of such concepts, as data, knowledge, the facts, etc.

The interpreter - the program which can carry out other programs. Unlike the compiler which only prepares programs for their further execution in other environment, the interpreter immediately starts performance of sequence of commands.

Cloning - manufacture of an identical copy on the basis of the initial sample.

Lexeme - a word considered as unit of dictionary structure of language in aggregate of all of its concrete grammatical forms, and also all possible values (semantic variants). In one lexeme different paradigmatic forms of one word (for example, « the dictionary, by the dictionary, to the dictionary » and т are united. Item) and the different semantic variants of a word depending on a context in which it is used (for example, "salt" in sense of the name of substance and in value of that gives an acuteness or interest to any statement, an idea and т. Item).

Lexicology - section of linguistics in which the dictionary structure of language is investigated, its lexicon, interaction between word meanings and concepts. Concepts are international more often, whereas word meanings национальны. Studies laws of functioning and development of dictionary structure of language.

Linguistics - a science about language. Object of linguistics is the structure, functioning and historical development of language, language in all volume of its properties and functions.

Mathematical linguistics - the mathematical discipline developing the formal device for the description of a structure of natural and some artificial languages. Includes theories and ways of the description of syntactic structures, formal грамматик and analytical models of language.

Melody - linear sequence of events. The melody represents the horizontal coordination of sounds. These sequences contain repeating images of sound objects (duration and tone).

Measure - function which establishes conformity between quantitative and qualitative characteristics of the phenomena, objects and processes. For example, it allows to coct such characteristics, as “volume”, “probability” “speed” and numerical sizes from some set of values.

Neuron - a specific cell which carries out perception, processing and transfer возбуждений from receptors to other neurons.

Nerve - connection between neurons on which are transferred excitation. Аксон - a unique shoot on which impulses are transferred to other cells. Синапсы - contacts on which impulses act inside of neuron.

Feedback - return influence of results of process on its course or operated process on operating body. Distinguish positive and negative feedback. Positive - if results of process strengthen it, negative - when results of process weaken its action.

Paradigm - system of forms of one word, reflecting modifications of a word on grammatical categories inherent in it, for example, on a sort, number and a case for nouns, to the person, time, a kind and other for verbs; the scheme of change of a word on grammatical categories; the sample of type of declination or conjugation.

Variable - one of the basic concepts of mathematics, logic and programming. In higher mathematics the variable is understood as some “size” which can “change”, accepting during this change various “values”. In the formalized languages of mathematical logic of a variable the symbols of the fixed kind able under certain conditions to be replaced to expressions refer to. In programming on a variable the three - the identifier, area of storage and the value connected with them is understood.

Polymorphism - in the physicist, mineralogy, chemistry ability of some substances to exist in conditions with various nuclear crystal structures. Distinction in structure causes also distinction in properties of polymorphic updating the given substance.

The concept - in formal logic is the elementary unit of cogitative activity possessing known integrity and stability and taken in derivation from verbal expression of this activity. The concept is that is expressed (or is designated) by any independent part of speech and if to pass from scales of language as a whole to “micro level” - that a sentence part.

The offer - the basic unit of the coherent speech characterized by certain semantic and structural attributes. Under the attitude speaking to an idea of the offer stated to them share on three groups: narrative, exclamatory and interrogative.

Process - consecutive change of conditions of system, as a result both internal, and external actions.

Ribosome - the end cellular particles which are carrying out biosynthesis of fiber; ribosome are found out in cells of all without exception of alive organisms: bacteria, plants and animals; each cell contains thousand or tens thousand ribosome. Fiber of all organisms consists of 20 kinds of amino acids. Each fiber is characterized by the certain assortment and a quantitative parity of amino acids.

Synchronization - procedure of the coordination of processes of processing or data transmission. Synchronization is carried out at a physical level: by means of the clocking setting the uniform standard of discrete time for management by process of signaling.

System - set of objects which can be allocated from the environment by means of border. The system can energetically or is information to cooperate with other systems.

Word - central unit of language. In any language there is a word meaning "word". The basic properties of a word: phonetic and semantic, reproducibility, syntactic independence, valency. Words share on significant (designating some concept) and service which serve for communication of words among themselves.

Word form - the word considered as the representative of a certain lexeme and the certain grammatic form. Set of all word forms of the given lexeme forms its paradigm.

The message - object by means of which there is an interaction. The message always has the physical nature and unlike all other phenomena, leads to change of a condition of system which has received this message.

Condition - a set of parameters and their values describing behaviour of system in time.

Environment - includes all systems and all objects which possess physical and information properties. It is possible to consider environment, as universal system which surrounds any system entering into it.

Energy - the general quantitative measure of movement and interaction of all kinds of a matter. In the classical physics, energy of system varies continuously and can accept any values. In the quantum theory, energy of a microsystem can accept only discrete values.

THE BIBLIOGRAPHY

1. **Aristotle.** The Basic Works of Aristotle. – New York : The Modern Library, 2001.

2. **Berners-Lee, T.** Information Management: A Proposal. – CERN, 1989. – May. (www.w3.org/history/1989/proposal.htm)

3. **Brooks, F.** The Mythical Man Month. – Boston: Addison Wesley, 1995.

4. **Bush, V.** As We May Think // Atlantic Monthly. – 1945. – Vol. 176. – № 1. – P. 101–108.

5. **Cherkassky, V., Mulier, F.** Learning from Data. – New York: John Wiley & Sons, 1998.

6. **Cohen, P. R., Fertig, S., Starr, K.** Dependencies of discourse structure on the modality of communication: telephone vs. teletype // Proceedings of the 20th conference on Association for Computational Linguistics. – Toronto, Ontario, Canada. – 1982. – June 16-18.

7. **Dijkstra, E. W.** The Structure of the THE Multiprogramming System // Communications of the ACM. – 1968. – Vol. 11. – № 5. – P. 341–346.

8. **Einstein, A.** Relativity. The Special and the General Theory. – New York: Three Rivers Press, 1961.

9. **Fellbaum, C.** WordNet: An Electronic Lexical Database. – Cambridge, MA: MIT Press, 1998.

10. **Feynman, R. P.** Feynman Lectures on Computation. – Cambridge, Mass.: Perseus Publishing, 1999.

11. **Gödel, K**. On Formally Undecidable Propositions of Principia Mathematica and Related Systems. – New York : Dover Publications, Inc., 1992.

12. **Heisenberg, W.** Physics and Philosophy. The Revolution in Modern Science. – New York : Prometheus Books, 1999.

13. **Heisenberg, W.** The Physical Principles of the Quantum Theory. – Leipzig, Dover Publications, Inc., 1949.

14. **Ingram, J.** The Notation of Time // Contact Magazine. – London, 1985.

15. **Maass, W., Natschlager, T., Markram, H.** Computational models for generic cortical microcircuits // Computational Neuroscience: A Comprehensive Approach / In J. Feng, editor. – Boca Raton: Chapman & Hall/CRC, 2004. – P. 575–605.

16. **Maass, W., Natschlager, T., Markram, H.** The “liquid computer”: A novel strategy for real time computing on time series // Special Issue on Foundations of Information Processing of TELEMATIK. – 2002. – Vol. 8. – № 1. – P. 39-43.

17. **Martin, J.** Cybercorp: The New Business Revolution. – 1996. – ISBN 0-814-40351-4.

18. **McCulloch, W., Pitts, W.** A logical calculus of the ideas immanent in nervous activity // Bulletin of Mathematical Biophysics. – 1943. – Vol. 7. – P. 115–133.

19. **Neumann, J.** The Computer and the Brain. – 2-nd edition. – New Haven, Conn.: Yale University Press, 2000.

20. **Neumann, J.** First Draft of a Report on the EDVAC: Contract 670-ORD4926, Philadelphia, PA. – Morse School of Electrical Engineering. University of Pennsylvania. – 30 June 1945.

21. **Pink, D. H.** A Whole New Mind. – New York: The Berkley Publishing Group, 2006. Gravity Analysis Domain // AI Magazine. – 2002. – Vol. 23. – № 4. – P. 51-68.

22. **Shannon, Claude E. and Weaver, Warren.** The Mathematical Theory of Communication – Urbana and Chicago: University of Illinois Press, 1949

23. **Shannon, C.** The Bandwagon // IRE Trans. Information Theory. – 1956. – Vol. 2.

24. **Stay, J. F.** HIPO and integrated program design // IBM Systems Journal. – 1976. – Vol. 15. – № 2. – P. 143.

25. **Stravinsky, I., Craft, R.** Conversations with Igor Stravinsky. – Berkeley: University of California Press, 1980.

26. **Tecuci, Gh., Boicu, M., Marcu, D.** Training and Using Disciple Agents: A Case Study in the Military Center of Gravity Analysis Domain // AI Magazine. — 2002. Vol. 23. № 4. P. 51_68.

27. **Teller, A., Veloso, M.** Neural programming and an internal reinforcement policy. In First International Conference Simulated Evolution and Learning. – Springer-Verlag, 1996.

28. **Turing, A. M.** Computable Numbers, With an Application to the Entscheidungsproblem // Proceedings of the London Mathematical Society. – 1936. – Vol. 42. – № 2. – P. 230–265.

29. **Watson, J. D., Crick, F. H. C.** Genetical implications of the structure of deoxyribonucleic acid // Nature 171. – 1953. – May.

30. **Wiener, N.** Cybernetics. – 2-nd edition. – Cambridge, Mass.: MIT Press, 1961.

31. **Winograd, T., Flores, F.** Understanding Computers and Cognition: A New Foundation for Design. – Addison Wesley, 1987.

32. **Заде, Л.** Понятие лингвистической переменной и его применение к принятию приближенных решений / Л. Заде. – М. : Мир, 1976.

33. **Жданов, А. А.** Бионический метод автономного адаптивного управления // От моделей поведения к искусственному интеллекту / Под ред. В. Г. Редько. – М. : УРСС, 2006.

34. **Колмогоров, А. Н.** Теория информации и теория алгоритмов / А. Н. Колмогоров. – М.: Наука, 1987.

35. **Раушенбах, Б. В.** Геометрия картины и зрительное восприятие / Б. В. Раушенбах. - СПб.: Азбука-классика, 2002.

36. **Раушенбах, Б. В**. Пространственные построения в древнерусской живописи / Б. В. Раушенбах. – М. : Наука, 1975.

37. **Сеченов, И. М.** Рефлексы головного мозга // Сеченов И. М., Павлов И. П., Введенский Н. Е. Физиология нервной системы: Избранные труды: Вып. 1 / Под общ. ред. К. М. Быкова – М. : Гос. изд-во мед. лит., 1952.

38. **Шеннон, К.** Работы по теории информации и кибернетике / К. Шеннон. – М. : Изд-во ин. лит., 1963.

INDEX

J

K

L

M

N

O

P

Q

R

CONTENT