STRESS TESTS: 14 OPTICAL CHARACTER READERS Page 220 APRIL 1991

BYTE

A McGRAW-HILL PUBLICATION

Taming A Sea of Paper
Page 156

An insider's look at computing in the troubled U.S.S.R.

Soviet Computing

Intel Speaks:
–the "Micro 2000" Project page 131
–a first look at a 100-MHz i486 page 27
Mac Video Revealed
Under SPARC's Hood
How to Network Windows 3.0
Pournelle's Annual Awards

PLUS
3 Laptop Pointing Devices
5 Object-Oriented Script Languages
ALR's Six i486 Multiprocessors
IBM's Speedy i486-Based Portable
GeoWorks:
Windows for the Rest of Us?
SPC's "InfoAlliance"

COVER STORY

FEATURE

Computing in the U.S.S.R.

Soviet "informatics," suffering from years of official policy that has hindered hardware and software development, looks toward the future

IGOR AGAMIRZIAN

On December 7, 1988, the academician Andrei Petrovich Ershov died in a Moscow hospital at the age of 57. His death went unnoticed in a country concerned with the tragic consequences of the Armenian earthquake. However, for specialists routinely dealing with computer science in their work, the event signified the end of an era.

This article is not an obituary of Ershov. It may, however, be the obituary of Soviet computer science, a demise that threatens to become the straw that breaks the back of our collapsing economy.

In the Beginning . . .

The first computers appeared in the late 1940s and early 1950s in the U.S., Great Britain, and the U.S.S.R. The cold war between the East and West caused an avalanche in the development of military equipment, and the creation of new arms required ever-increasing calculations. Thus, the first generation of computers was intended to solve the problems of ballistics and nuclear physics.

In the 1950s, however, the U.S. began using computers to solve business problems. Work began to automate programming and create high-level languages; programming in machine codes had become too time- and money-consuming. The first valve-operated monsters were replaced with solid-state devices. Backing, main, and internal storage volumes grew significantly. Performance became thousands of times faster.

Progress in the development of high-level programming languages and their compilers predetermined the look of programming. In the 1960s, ALGOL occupied the leading role in the U.S.S.R. Classical Soviet developments of compilers were associated then with ALGOL.

A series of computers called BASM (the Russian abbreviation for large electronic computing machines) was created under the guidance of S. A. Lebedev. In the mid-1960s, this led to the construction of the first line of Soviet program-compatible computers, called the M-20. The family included the M-20, BASM-3M, BASM-4, M-220, and M-222. The first compilers engineered for these computers were called Alpha, TA-1, and TA-2. Ershov, S. S. Lavrov, and M. R. Shoura-Boura headed teams of programmers that created the first Soviet ALGOL compilers for the family of M-20 computers.

In the late 1950s and early 1960s, the U.S.S.R. established centers for the

ILLUSTRATION: JOHN BREAKEY © 1991

development of informatics and computer science. (Informatics deals with information processing, including mass media, publishing, and intelligence activities. Here, however, I use the term to mean a sphere of problems associated with data processing involving the use of technical aids.)

In the U.S., the motivation for developments in computer science shifted from military applications to the search for methods of increasing labor productivity. Advances in microelectronics led to the creation of ICs, which permitted the development of a third generation of computers. Software became more complicated, and operating systems replaced the second-generation master-control programs.

At the same time, informatics and computer science in the U.S.S.R. again came under the influence of politics. (The golden age of Soviet informatics fell within the years of Khrushchev's thaw—you cannot imagine cosmonautics without computer science, and the attention Khrushchev paid to cosmonautics is well known.) Conservative forces that came to power were interested only in preserving the status quo and unavoidably contributed conservatism to policies regarding technology. Developments were curtailed all over the country. The American IBM Model 360 (1965) was suddenly adopted as the Unified Computer System by the Council for Mutual Economic Assistance (CMEA), or Eastern European countries.

The era of developments ended in 1968 with the appearance of the last computer in the BASM family, the BASM-6. After that, there was nothing. Millions of rubles were invested in the development of computer science to no end. The country once again took the fruitless road of copying Western models and assumed that by copying another technical innovation it would save so much that the products would surpass those of competitors. (Edsger Dijkstra, a classic software engineer who visited the U.S.S.R. in the late 1970s, said in a public speech delivered in the Grand Hall of the Academy of Sciences in Leningrad that he regarded the fact that the U.S.S.R. produced IBM computers as the biggest U.S. victory in the cold war.)

You might think that the history of informatics in the U.S.S.R. after 1965 is that of a thoroughly planned strategy against the Soviet people. But this strategy was planned not by bad Americans but by good and experienced Soviet leaders trying to improve the public welfare. The cause was indifference, selfishness, apathy toward tomorrow's problems, an absence of responsibility to the people—everything associated with what we now call the administrative system.

That system failed not only in informatics and computer science but in agriculture, industry, transportation, and communications. One example: Why did the shuttle *Buran* take off so late? Undoubtedly, one reason was the absence of the computers required to simulate its aerodynamics. Americans made calculations for their shuttle on computers of the 1970s, which surpassed Soviet computers of the 1980s.

The Soviet Seventies

In the early 1970s, the most popular computers in the U.S.S.R. were the M-222 and the Minsk-32. Scientists at the Nuclear Physics Institute in Dubna created one of the first Soviet FORTRAN compilers, and FORTRAN for the M-20 appeared. A new line of Soviet computers was announced—the Elbrus—whose design was suspiciously similar to that of systems from the American company Burroughs. Everyone looked forward to the appearance of ES ("Unified System") computers, which were to be compatible with the IBM Model 360.

I first saw an ES computer when I was studying informatics and computer science at the faculty of mathematics and mechanics at Leningrad University. My fellow students and I were

An American Programmer in Moscow

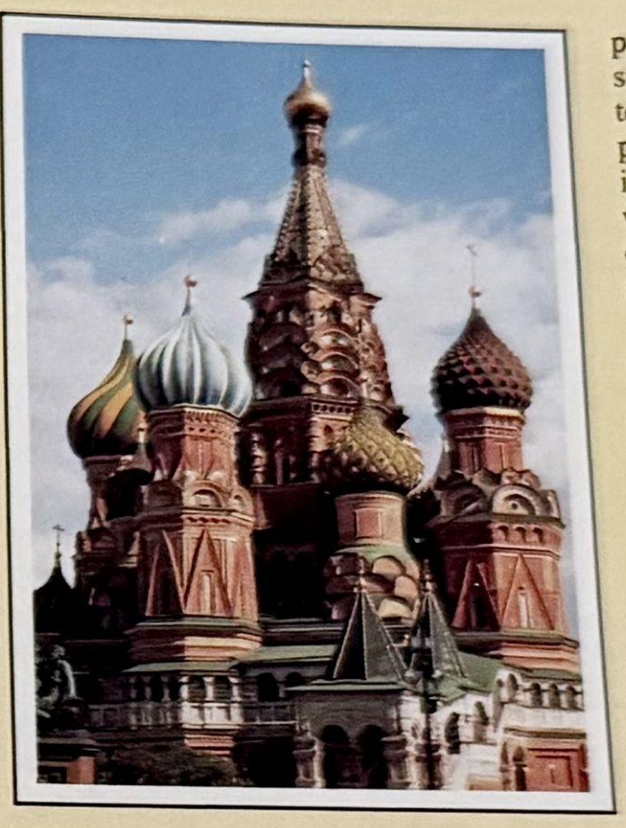
Alex Lane

Nearly 15 years had passed since I had been in the U.S.S.R., and the border guards were just as steely-eyed as ever. But I passed through the line with hardly a glance from the customs officer, met my driver, and headed for downtown Moscow. For the next few days, as part of the Borland team at the First International Computer Forum, I'd be showing off Borland products, talking with Soviet programmers, and seeing Soviet computing up close.

It should come as no surprise that Soviet programmers look and act very much like their American counterparts. They ask much the same questions, too. While waiting for a customs official to unlock a vault, several young programmers who had volunteered to help carry boxes of literature plied me with questions. What languages did I program in? What did I think of FORTRAN? What about upcoming Borland products? Windows? OS/2?

Not all questions were about computing. One fellow asked me how Americans went about finding jobs after graduation. My answer fascinated them, since the idea of going out on your own to look for work is quite foreign to the Soviet mind. Another naively asked if Americans really were taught to hate Russians.

When it comes to producing code, I found that Soviet programmers know their stuff. Their major stumbling block isn't the quality of hardware or software; it is the quantity. I saw mostly 286 and 386 hardware, obtained from Pacific Rim countries. Software is routinely pirated, and masses of accompanying documentation are no barrier to piracy, despite the nonavailability of photocopy machines. Soviet programmers simply combine computer technology with the *samizdat* (self-publishing) technique of political dissidents to create complete versions of documentation for electronic distribution.



Ironically, copy-protection schemes are high on the list of software that Soviet programmers like to develop. I sat through several demonstrations of such schemes, complete with vague explanations of the modus operandi, and none of the programmers could be convinced that the schemes could be broken.

Closely associated with copy-protection software is work on antivirus programs. I was told viruses were a serious problem in the Soviet Union, especially since they were often aimed at "the system" by disgruntled individuals. One programmer was hard at work developing a virus-description language that would allow a programmer to easily create software to detect and neutralize newly identified viruses.

While most Soviet programmers understand English and work with English source code, a third major area of programmer involvement is the Russification of foreign programs. Efforts here range from some neat hacks by one-person shops to complete systems, such as the one developed by the ParaGraph joint venture for Paradox.

Of course, programmers everywhere dream of developing the next killer application—the one that will sell millions of copies and bring them fame and fortune. Soviet programmers are no different, and for the duration of the forum, there was a steady stream of hopefuls at the Borland booth wanting to show their pet applications. Technically, most of the programs were pretty sharp. However, virtually all of them had been developed with no market in mind and so were of limited interest.

Soviet computing is on the verge of an expansion akin to the growth in the U.S. in the 1980s. Soviet developers are becoming aware of the domestic and international markets. Don't be surprised to find the U.S.S.R. emerging in this decade as a major player in the computer revolution.

Alex Lane is a senior technical writer for Borland International. He can be reached on BIX as "a.lane."

proud of Soviet software engineering, convinced that there were positive aspects in our dated computer hardware: It was a training factor that enabled us to engineer applications that were better than American programs. Americans, we thought, did not care about efficiency, but we had to, so we were better programmers. (It was true, to a certain degree. At least, the West always treated Soviet programming schools with respect, and a Soviet programmer emigrating to the West could find a job with ease.)

At that time, programming was becoming a mass specialty in our country, and the need for automatic-control equipment for thousands of plants required new software. As the ES computers were put into use, more users preferred FORTRAN and PL/I, and the ALGOL traditions in the U.S.S.R. faded away.

continued

System programming became devalued, as Soviet clones of foreign computers and "borrowed" copyright software became available.

Few people realized that further progress in informatics was impossible without system developments. We naively hoped that small groups of highly qualified software engineers scattered all over the country would be able to withstand the powerful stream of copyrighted software. We did not understand that the creation of successful software is another science, for which understanding optimum translation algorithms is insufficient.

The late 1970s saw the development of CM computers, a family of small computers, bringing a CMEA program for the creation of computers modeled after the most successful American minicomputer, the PDP-11 from DEC. As the system's mass production was organized, the country soon filled with new Western software. There followed a period of conversion (i.e., the adjustment of Western software to Soviet computers, involving the translation of program messages into Russian.)

Soon, we forgot our own system development traditions. It was much more convenient and profitable to steal than to create something of your own. A whole generation of programmers was unable to create its own programs but could readily understand and improve other programs.

How could borrowing (or more honestly, stealing) foreign intellectual efforts become almost a state policy for informatics and computer science? The reason lies in the deepest contempt for intellectual creative work. In developed countries, copyright laws protect the humanitarian as well as scientific and technological spheres. In the U.S.S.R., the product of intellectual work does not belong to its author; it does not even belong to the organization within which it is developed.

Unfortunately, this is one of the main reasons for our decline in these disciplines. Why should we invest in products if we can obtain everything free? At first glance, it might seem more profitable to use stolen programs. Most Soviet software is only produced in a few copies, and some programs are used only by their authors or programming organizations. A special fund of algorithms and programs, state and regional, has existed for many years in the U.S.S.R. Programmers are well aware of the complications they must overcome to incorporate software into this fund.

Some readers may object to my thesis that the cause of all mishaps in Soviet informatics and computer science lies in copying. What about the international tendency toward unification of computer architectures and software? Clones of the Intel 8086 chip are produced not only in the Soviet Union (the KM1810BM86) but also in Japan, Hong Kong, and Taiwan. IBM PC–compatible computers are produced not only in the Soviet Union (where they are called Iskra-1030 and ES-1840) but by hundreds of companies in dozens of other countries.

But there are differences: Unlike companies in the U.S.S.R., other countries' companies buy licenses and technologies. Buying technologies gives them a lead in time (Southeast Asia begins to copy new American products within several months, while the U.S.S.R. waits several years) and quality (the reliability of Taiwan-made microcircuits surpasses that of masterpieces produced by the Soviet Ministry of Electronic Industry).

Long ago the U.S.S.R. brought itself into information dependence on the U.S. In the 1970s and 1980s, there was little support for the rare attempts at independent hardware and software development. The absence of competition led to the loss of objective quality criteria and the attempt to improve the product.

There is a saying among Soviet programmers: Programs may be bad, good, or working. We like good programs very much. Americans prefer working programs. Because the working

programs and hardware are useful, the Soviet dependence on American information technologies grew, and that dependence became marked when personal computers appeared on the scene.

Bypassed by the Personal Computer Revolution

With the advent of personal computers, a revolution broke out in the world, but it bypassed the U.S.S.R. Its leaders noticed nothing. Soviet industries, infamous for their immobility and habit of copying foreign models, failed to produce decent computers. Even substandard computers were manufactured in such small lots that you could not speak seriously about the computerization of the country.

Alarm was raised. Many famous specialists expressed apprehension and offered constructive steps. Legislative acts were issued, new departments were formed, production plans were set up and failed. Ershov, who was well aware of the danger of further decline in informatics and computer science, put forward a slogan: "Programming is the second literacy."

The personal computer revolution bypassed the U.S.S.R.

That slogan did the trick: It drew attention to the problems of computerization and education in informatics. A vast educational program was started, and informatics became an obligatory school subject. However, the pioneers of this process assumed that industry would provide schools with the necessary equipment within two to three years. Industry failed, and in most schools, the subject is taught on paper only. Students are lucky if they are able just to *look* at a computer two or three times a year. Thus, the students become indifferent, or they do not like the subject. In many cases, the teachers set the tone: Informatics and computer science are usually presented by teachers of physics or mathematics who often have only vague knowledge of the subject.

In a paper published in an American magazine several years ago, comparative figures were given on computerization in the U.S., Japan, Great Britain, and other countries. The author was disturbed that the U.S. did not occupy the leading position in the computerization of education. "We are lagging behind, which may lead to a tragic end," the author concluded—a funny and sad conclusion for the Soviet reader. The U.S.S.R. is not just lagging behind; it is facing the risk that a generation of Soviet citizens will be unable to understand citizens of all but the least developed countries. We Soviets risk separating ourselves with a new curtain—not an iron curtain this time, but a steel curtain of ignorance. We risk finding ourselves alone, because even our CMEA partners are ahead of us by more than two or three years (computerization in Poland, Hungary, and Bulgaria—although they do have certain problems—is decent enough and is improving much faster than the U.S.S.R.'s).

A Few Hopeful Signs

In recent years, some positive tendencies appeared. Unfortunately, all processes are too slow, and we lack time. Solutions to certain problems are not efficient and often hinder, rather than accelerate, resolution of the problem. However, fine efforts exist, such as the temporary science and technology team, Start, set up in 1985 to design a new generation of hardware and software. The result of this effort is a working model of an original and very promising multiprocessor computer and accompanying original software. Unfortunately, the project ended in spring 1988, and our industries seem to be uninterested in the results of this development.

Nor has there been much interest in the Soviet Academy of Sciences' project Shkola (School), headed by E. P. Velikhov. Shkola was a serious attempt to solve the problems of computerization. Regrettably, no progress has been made. That cannot happen until the national economy is restructured.

But most Soviet enterprises, even in modern spheres like microelectronics, are not interested in developing and improving their production capability. Unfortunately, developments in informatics and computer science require powerful material, information, and technological bases; and it is useless to hope that new Soviet computers may—even if the developers enjoy the most favorable conditions—be created in university laboratories or institutes of the Academy of Sciences.

When a successful model is made, there arises a need for large-scale ICs, whose design requires equipment the U.S.S.R. doesn't have. We cannot obtain CAD systems as our foreign counterparts do because we do not have adequate technical resources—that is, computers and peripherals (e.g., high-resolution displays). These systems belong to enterprises that are uninterested in the introduction of new products. The vicious circle is closed. It may be opened again only as a result of deep restructuring of the economy.

The beginnings of economic self-regulation (a socialist market) is the only chance to catch up in informatics. However, the market mechanism cannot correct the decline. Given the existing (or more accurately, nonexisting) copyright laws, it is more profitable for enterprises to copy products or to buy licenses for production from abroad, primarily from the U.S.

Therefore, together with economic changes (here, let me be a little naive and expect that the ruble will be convertible in two to three years), we urgently need state stimulation of promising areas of science and technology. The state should create economic conditions under which the use of domestic products would be more profitable than the use of foreign products. This may be achieved by a combination of actions, including judicial, financial, and tax measures. It should be done as quickly as possible; in fact, it should have been done yesterday.

While the creation of hardware requires considerable resources, personal computers greatly reduce such requirements for software. Unfortunately, the U.S.S.R. suffers from a deficit of equipment, and its prices are 150 times higher than those of the U.S. (neglecting the symbolic exchange rate and considering real wages, the state price for a professional personal computer is about 40,000 to 50,000 rubles, while a qualified programmer earns about 300 rubles a month).

This was why programming and mediator cooperatives and Centers of Scientific and Technological Creative Work of the Youth were created for the development of software products. It is curious that mediator organizations, although charging up to 50 percent of the contract cost for establishing relations

An Editor's View

Last year, BYTE's editor in chief, Fred Langa, was a guest at the First International Computer Forum in Moscow. He later attended the first East-West High-Tech Forum in Budapest, Hungary. Here are some of his impressions of Moscow and Soviet computing.

Moscow is a monochrome city, built primarily of beige bricks, with some concrete and red brick here and there. The city is amazing for what is there, and equally amazing for what is not there.

Looking out the window of my hotel, I could see four TV antennas, and that was all. There were no satellite dishes anywhere to be seen, and I saw only one pair of microwave dishes the whole time I was there.

In the shops, there were few cash registers. Some of them had a cash drawer and did all their accounting manually on a piece of paper. The high-volume shops—such as food vendors and theaters—used an abacus. It was ironic that one of the forum's themes was the need to lift the Coordination Committee on Export Controls restrictions so more advanced machines (including RISC machines) could be imported into the Soviet Union—this is akin to having a donkey and wanting it to go faster, so you feed it unleaded high-test gasoline.

It is simply not going to work. What is needed are basic calculators and simple computers in large numbers. That seems to be far more important in terms of fundamentally changing Soviet society than getting a very limited number of workstations into the hands of an elite. However, anything that brings more computers in and enhances a computer culture—with all the intercommunication and worldwide standards that implies—would help.

At the forum, Jack Byers, vice director of JV Dialogue, spoke of the many difficulties that joint ventures face. Each joint venture in the U.S.S.R. has to have a wide base of support: hotels, so you have a place to put international guests; food, so your workers won't have to stand in line at restaurants or food stores; groceries, so your workers will not have to worry about getting essential items for their homes; cars, to move workers around; gasoline and mechanics for the cars—the list goes on and on. A successful joint venture doesn't just provide a job or an opportunity; it has to build an entire minisociety for its workers. You can't take anything for granted.

Adventures in Soviet Software

The East-West High-Tech Forum was a kind of "Dating Game" for would-be partners in joint East-West high-tech ventures. Among the many attendees of the forum was Rusty Schweickart, president of NRS Communications. (You may remember him from his astronaut days.) Schweickart noted that computers in the U.S.S.R. are status symbols and are held tightly by those few who have them. A Soviet citizen added that it was like paper clips: Only 5 percent are used to clip papers; the other 95 percent are used in unexpected ways. In the same way, computers get used in very different ways. They are so flexible and valuable that they're snapped up and jealously guarded.

Irina Savelyava, a Soviet lawyer, told me that Hungary was the first Eastern-bloc nation to offer software copyright protection. The Soviet Union is one of the last remaining countries *without* such protection. A long discussion on intellectual property ensued, during which the Soviets' consensus was, "There is no piracy in the Soviet Union, because it is not illegal to copy software." This is not a trivial point; the U.S.S.R. has no history, mind-set, or legal basis for the private ownership of software. (They're even struggling with ownership of concrete items, such as land and factories. Ownership of intangibles like software is hard for them to understand.)

Stepan Pachikov, general director of ParaGraph, perhaps the most successful Soviet software company, says he runs into this issue every day: People buy one copy of a ParaGraph package and then call to ask for instructions on how to make copies for the other 100 or 1000 people in their firm. The idea of one-copy-one-user or a site license is unheard of.

Nevertheless, Pachikov maintains that "free copying" (not piracy) is good because it seeds the market and helps develop users and programmers who otherwise would have no access to Western software. Case in point: A software package that costs $400 in the West costs about *two years' wages* in the Soviet Union. Can you imagine spending two years' salary on, say, Microsoft Word? As Pachikov said, "Gorbachev makes 24,000 rubles per year, or about $1200. Gorbachev himself could only afford to buy three copies of Microsoft Word in a year."

In the West, product support is often used to discourage piracy: no registration number, no support. But in the East, says Pachikov, vendors really can't offer good support (although some are trying) because "it's harder to get a phone line than to get a nuclear submarine."

Hardware Hassles

Leonid Tomberd, a researcher at Estonia's Institute of Cybernetics, related some problems with the development of Soviet hardware. For instance, most U.S. chip leads are spaced $\frac{1}{10}$ inch apart. The Soviet ministry in charge of cloning Western chips mandated metric spacing, but $\frac{1}{10}$ inch works out to be about 0.254 millimeters, an odd metric size.

The Soviet solution? A *metric inch*—0.25-mm spacing. This means that Soviet clone chips can be exact electrical and functional equivalents of their Western counterparts, and look exactly the same—until you try to plug them into a Western socket. They won't fit.

That's more than an inconvenience. It means all Soviet chip assembly hardware won't work with anything but Soviet chips, and a fair chunk of the actual Soviet chip-fabrication equipment and its products are worthless on world markets.

To make their quotas, Soviet chip manufacturers pump out chips with little or no quality control. Tomberd says a typical DRAM shipment has 25 percent to 50 percent bad chips. For CPUs, he said, it's not uncommon to get a defect list with each chip—a list of instructions that chip will not run. Can you imagine trying to assemble a system from chips like that? Can you imagine programming for chips like that?

To force support for the Soviet computer industry, the government has mandated the use of Soviet equipment. For example, Tomberd says that ministries were forced to use Soviet IBM mainframe clones that cost 1 million rubles, took up precious office space, and required 30 people to operate and maintain—yet had a throughput roughly equal to about two Western personal computers.

Even now, the ministries can't get rid of these dud mainframes. No one wants them; no one will pay anything even close to the paper value of the machines. If they give them away or sell them at their real value, it will look as though the sellers are doing something illegal—dumping a machine supposedly worth a million rubles for virtually nothing. Until the laws of depreciation change, they're stuck with the old gear. Once again, there's no correlation between real worth and actual prices, and they have no easy way out.

between a customer and a developer, found themselves in very favorable conditions.

Currently in the economy, such mediators are necessary. However, in the future, programming should be recognized as a free, creative trade, and the organizations should be able to conclude direct contracts with the programmers. (This is possible today, but the job is paid from the organization's wage fund, and this is highly unprofitable for the organization.)

A Soviet ACM

In the U.S., public organizations like the Association of Computer Machinery have had a great effect on computer science. The absence of such an organization in the U.S.S.R. has hurt the development of informatics and computer science. The functions of such an organization were often borne by state working groups and committees of the Academy of Sciences, especially the Commission for System and Mathematical Software for Computers (the so-called Ershov's commission, headed by Andrei Ershov). Such organizations could deal with scientific and technical problems, but not with the social and judicial aspects of informatics development. Until recently, the creation of a public organization that could deal with all the existing problems was impossible, because such an organization could threaten the departments' monopolies.

On February 17, 1989, the All Union Society of Informatics and Computer Science was convened. About 200 voting delegates and 700 guests gathered for the inaugural congress and were handed the new society's draft charter (anonymous, as usual). Unfortunately, its authors thought traditionally; the charter did not address problems like the protection of authors' rights, public examinations, the organization of an information exchange, and so on. The only new idea was the creation of self-supporting Centers of Scientific and Technical Services run by the society; that is, another group of mediator organizations. The society was to be a consultative body of the State Committee for Informatics and Computer Science of the U.S.S.R., the same body that had initiated the formation of the society. The state committee had decided to create a society in its pocket, and the draft charter left departmental interests associated with informatics and computer science intact.

The absence of openness and democracy at the congress resulted in an organization that was composed of delegates who had no idea how or when they were elected to their positions. I. N. Bukreev, a deputy chairman of the state committee, who is now the chairman of the presidium of the society, told one of the congress delegates, "You were elected by voting secret from you!" What resulted was a congress made up of delegates representing the leadership of large enterprises, institutions, or ministerial departments, while the areas of science and technology were poorly represented. Some of the leading scientists in informatics and computer science were not invited, while, as the report of the credentials committee stated, there were 14 delegates without higher education.

Time will show the vitality and usefulness of the All-Union Society for Informatics and Computer Science. I will be glad if it improves informatics and computer science in the U.S.S.R. For now, it seems that users groups—which in the 1970s played a significant role in spreading information, forming public opinion, and evaluating new software products—might be more useful. Perhaps they should take on another form, but public initiative is necessary to ensure consolidation of forces at the upper level. Some timid, positive attempts have been made, such as the creation of clubs of professional programmers in Leningrad and other cities, which may be the basis for uniting efforts by the leading qualified developers.

This article contains words of sorrow. The current state of Soviet informatics is not happy, and there are no promising perspectives for tomorrow. Those involved in informatics and computer science can only look to the future for some hope. ■

Igor Agamirzian received his undergraduate and doctoral degrees at Leningrad University. He is currently a senior research fellow at the Leningrad Institute of Informatics of the Soviet Academy of Sciences, as well as an assistant professor for the computer science department at the Electrical Engineering Institute, Leningrad. You can reach him on BIX c/o "editors."

BYTE

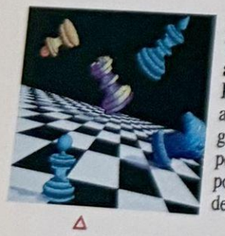
April 1991 Volume 16, Number 4

COVER STORY

Computing in the U.S.S.R.

PAGE 120

NEWS



FEATURES

STATE OF THE ART

REVIEWS

COVER ILLUSTRATION: JEAN BAILEY © 1991

HANDS ON

BYTE (ISSN 0360-5280/91) is published monthly with an additional issue in October by McGraw-Hill, Inc. U.S. subscriber rate $29.95 per year. In Canada and Mexico, $34.95 per year. Single copies $3.50 in the U.S., $4.50 in Canada. Executive, Editorial, Circulation, and Advertising Offices: One Phoenix Mill Lane, Peterborough, NH 03458. Second-class postage paid at Peterborough, NH, and additional mailing offices. Postage paid at Winnipeg, Manitoba. Registration number 9321. Printed in the United States of America. **Postmaster:** Send address changes, USPS Form 3579, and fulfillment questions to BYTE Subscriptions, P.O. Box 551, Hightstown, NJ 08520.

OPINIONS

INSIDE BYTE