**Speech at the occasion of the reception of the ITAC-NSERC price(January 26, 1989)**

Gregor v. Bochmann

Note: This was the **ITAC Award** (for academic excellence in research in the field of information technology) which was given by ITAC (Information Technology Association of Canada, an industry association) in collaboration with NSERC (Natural Sciences and Engineering Research Council of Canada) - this was the first time the award was given.

Bonsoir — good evening:

I am very touched to be honoured with this prestigious prize. Iwould like to thank the Information Technology Association ofCanada (ITAG) and the Natural Sciences and Engineering ResearchCouncil (NSERC) for the trust that such a distinction implies. I hope that I will be able to continue in the future to provide useful contributions to the development of information technology in Canada.

In the following minutes, I would like to give you an overview ofmy research area and the questions we are trying to answer. Iwould also like to make some observations about some issuesrelated to my research environment.

(The context: Communication protocols)

My research is related to what is called "communication protocols". You all know that computers are used for complex dataprocessing applications, and that often computers areinterconnected through various networks which allows them toexchange data between one another.Communication protocols are those rules which must be followed bythe interconnected computers for the exchange of data between oneanother. It is like a language used by people for the exchange ofinformation and ideas. If I gave this speech in Japanese, most ofyou would not understand my message because Japanese is not yourlanguage. Similarly, the designers of computer networks create a number of rules that must be followed by the computers duringtheir "conversation". These rules allow the computer to interpretthe bits, that is, the zeroes and ones, which are exchanged overthe transmission links. If a computer does not follow all therules of the communication protocol, it will not be able tocommunicate correctly with other computers using that protocol.One usually says that the computer is not "compatible" with the other machines.

The problems with protocols are that first, they are difficult todesign, and second, they are often difficult to implement. Onemust be very careful to avoid any design errors in the definition of the protocol. For example, if two computers that are exchangingmessages, enter a state in which each of them is waiting for theother to "speak", they may continue to wait politely forever. Inhuman conversation, nobody would wait indefinitely for an answer but computers are not that intelligent, they follow the programwhich was written as an implementation of the protocol.

The second problem, related to implementation process, comes from the fact that implementing a program that executes a particularfunction, in general, is difficult. Most, protocols are implemented in software programs that direct the computer to exchange dataover a network and to follow the relevant communication protocolduring these activities. It. is very common that these complex computer programs contain a few errors, and such errors may leadto a behavior which is contrary to the defined communicationprotocols. This in turns leads to incompatibility and lack of communication. In recent years, much attention has been given bythe data communications community to what is called "conformancetesting". This means that a given protocol implementation istested in order to verify whether it follows all the rules definedin the protocol specification.

It is important to note that these difficulties are not onlyencountered in the design and implementation of communicationprotocols, but also, to a perhaps lesser degree, in softwareengineering in general. In the past 10 to 20 years, there has beenincreasing discussion of the so-called "software crisis". This crisis concerns the constantly growing cost of the programs for most computer applications, compared to the cost of the hardware, which has always been decreasing. Although progress has been madein the past for improving the methods for software development, noimportant breakthrough is visible. While some mass-producedprograms are sold at low prices on the consumer market, the development of new programs is still difficult and expensive andthe increasing complexity of some more recent applications doesnot make the task easier.

(My research area :)

My area of research - and 1 am not alone in this field - is thedevelopment of formal description techniques or languages whichcan be used to precisely define communication protocols and othersoftware systems. Similar to programming languages, suchspecification languages can be processed and understood bycomputer programs much easier than English or French. There aretwo main advantages for using a formal language to definecommunication protocols, or specifications of software systems ingeneral*:*

First, there are no ambiguities in the interpretation of the specifications, since the language definition gives a precisemeaning to any valid specification that can be written in thelanguage. Traditional specifications, usually written in naturallanguage, such as English, usually contain some statements whichallow for different interpretations. This may result in systems that do not work correctly. Imagine for instance that a givenprotocol specification is implemented on two different computersby two different programming teams, the two implementations maybecome incompatible to one another due to different interpretations of the protocol specifications.

The second advantage of formal specifications is that they can be processed by computer tools which allow a partial automation ofthe protocol development process.There is, however, also a drawback for the use of formalspecifications: they are not immediately understandable by most people, compared to English, for instance. It is thereforenecessary that designers and implementers learn the language andthe associated tools, for taking advantage of their possibilities.

All of this will improve the software development life cycle,through "re-usability" and "fewer errors". Re-usability reducesthe cost of program development by using an existing module tosatisfy the requirements of a new application- Fewer errors alsoresult from the use of formal specifications and associated design, implementation and testing tools. These methods have aparticular impact on computer applications where errors must be strictly avoided, such as in high-reliability software for controlling real-time systems, for instance aircrafts or nuclear power stations.

(Mycontributions: )

As mentioned earlier, there have been no major breakthroughs inthis research area for some time now. It seems that advancementsare made in little steps, like constructing a building by addingone brick after the other. My contribution in this area has beento add some of these bricks, complementing the work done by otherresearchers in the field. For instance, it seems that some of mypublications in the late seventies have been the starting pointfor the development of the new research area on formalspecifications for communication protocols. During several yearsuntil 1985, I have played a key role in the development of the OSIspecification language called Estelle, which became aninternational standard in 1988. Also several protocolimplementation and testing projects performed in my research groupat the Université de Montreal have been in the forefront ofdevelopment.

(My own research orientation;)

My research orientation has been to combine theoretical resultswith practical applications, and I expect to continue along thatline. This is somehow related to the approach of physics, thefield in which Imade my PhD, where one tries to find a theorythat explains the multitudes of objects and processes which can befound in this world. Among two theories that explain the samethings, a physicist will usually give preference to the simpler one. Similarly in computer science, for the multitude of practicalapplications that are to be built, it is preferable to use languages and tools which have a simple structure andstraightforward theoretical foundation. The reasons are: ease ofuse and – may be – intellectual beauty.

(Comments on research environment :)

I would now like to make some comments on my research environment.First, I would like to say that I appreciate the liberty of actionconcerning research which I enjoy in my position as a professor at the university. I think that this has had a beneficial effect onmy research results. Another important consideration is financialsupport, for basic and targeted research. I appreciate very muchthe freedom with which I can use funds of NSERC's operating grants for various projects, sometimes not foreseen at the time when theresearch proposal was written. I consider it important to keep thebureaucracy associated with research projects to a minimum, especially in the area of basic research.

In addition to doing basic research, I find it important to get involved in application-oriented research projects, because theyprovide a perspective for the usefulness of the basic researchresults. Computer science is largely am engineering discipline and therefore, the application of research results is a mainobjective.

Presently, there seems to be a trend for stronger industry-universitycollaboration. I think this is a healthy development,especially in computer science, and should be encouraged. Suchcollaboration should include close contacts between theresearchers from the university and industry environments. Thiscould be obtained through research projects involving research personnel from university and industry, and through joint meetingsincluding seminars and discussions on particular research topics.

(Language issues:)

The fact that I represent a francophone university is important to me, even if French is not my mother tongue. I have been involvedfrom time to time in discussions concerning the quality ofresearch in French and English institutions in Canada. I thinkthat much excellent research is done at francophone institutions,in spite of the fact that francophone researchers have additionaldifficulties to overcome. These difficulties are simply based onthe fact that the working language is not English - English which is nowadays the main language for international communication.

First, of all, these researchers have to learn a foreign languageto do research. Second, depending on their fluency of English,they may have difficulties in expressing their research results,which may result in less recognition, additional difficulties forhaving papers accepted for publication, or research grantproposals well understood by the reviewers. The third difficultyis related to international exchanges, which are important forworld-class research. There are far fewer researchers, worldwide,that speak French compared to English. I have experienced severalcases where a bright foreign graduate student, or a researcherlookingfor a place to visit for a longer period, did not chose tocome to the University of Montreal because they did not speakFrench and were not interested to learn it.

I think it is important that English-Canadians realize that French is not just another language spoken in some parts of Canada, but that the use of a non-English language implies a number ofdifficulties that must be handled by French-Canadians. In order tosolve this problem for myself, I could have moved to an English university. But I do not think that this is a solution in general. Instead, a French research environment must be actively supportedin Canada, if one is serious about bilingualism. At the same time, it is important that the French-Canadian research community bewell integrated into the Canadian and international context.

( Final remarks: )

Before closing, I would like to stress that I have not been alonein doing the research for which I received this prize. I amgrateful for many fruitful discussions and intellectual exchangesfor which I would like to thank my colleagues, collaborators and students at the Université de Montreal, and many other researchersin universities, industries and government organisations. I wouldalso like to acknowledge the comprehension and support I receivedfrom department chairmen and the university administration.

Information technology is a fascinating field. There are fasttechnological advances and a multitude of different applications.I like to work in this area, and am looking forward to stayinginvolved in its future development.

Thank you for your attention.