

# PROTOCOL TESTING AND VERIFICATION WITHIN AT&T

Alfred V. Aho, Barry S. Bosik, and Stephen J. Griesmer

**Alfred V. Aho** is director of the AT&T Bell Laboratories Computing Science Research Center in Murray Hill, New Jersey. He is responsible for basic research in computer science. Mr. Aho, who joined the company in 1966, has a B.A.Sc. in engineering physics from the University of Toronto, and a Ph.D. in electrical engineering and computer science from Princeton University. **Barry S. Bosik** heads the Network Interfaces and Standards Department at AT&T Bell Laboratories in Holmdel, New Jersey, that tests customer premises equipment for conformance to AT&T network services' and products' interfaces. Mr. Bosik received a B.S. from the City College of New York, an M.S. from Princeton University, and a Ph.D. from Columbia University, all in electrical engineering. He joined the company in 1968. **Stephen J. Griesmer** joined the (continued on page 6)

This issue of the *AT&T Technical Journal* emphasizes the importance to AT&T of industry-standard protocols in the design of communications networks. The articles describe the broad range of work conducted within the company in the development of procedures and tools to verify and test communications protocols.

As Larry Bernstein noted in the preface, computer and communication networks are becoming more sophisticated, more pervasive, and more heterogeneous. Users want to interconnect equipment and software from different vendors to form seamless networks of diverse components that work reliably and efficiently. They want to manage their networks easily and, most importantly, to add new components and replace existing ones without making their previous investments obsolete.

To satisfy these user requirements, computer vendors and communication providers need to build products that can communicate with multiple networks and with other vendors' products, not just with one network or with the provider's own products. The ante has been raised: Computer and communications products will be evaluated on whether they conform to industry-standard specifications, pass appropriate conformance tests, and provide interoperability in a multi-vendor environment.

The translation of these user requirements to products is realized through communications protocols. A protocol is a set of rules or conventions by which one component or part of a network communicates with another. The creation of robust communications protocols is a formidable task because all modes of network behavior, including the failure modes, must be considered in the specification. Many industry-standard protocols for connecting equipment to communications networks have been defined; many new protocols are being defined for peer-to-peer equipment communication. Old standards are evolving as the sophistication of computer networks increases, and new standards are being drafted to create additional functionality and better performance.

Because robust protocols are vital to reliable communication networks, it has become necessary to create several kinds of software and hardware tools to analyze and test protocols. After a specification

---

for a protocol has been created, it is important to check that the specification properly provides all the required capabilities, and does not have undesirable properties such as deadlocking or preventing some users from achieving the network performance they expect. The process of analyzing a specification to determine its logical properties is called *protocol verification*.

After a protocol has been specified satisfactorily, it is implemented in software and/or hardware. Automated tools are essential to test that an implemented protocol does what its specification says; this process is called *conformance testing*. A thorough conformance test puts an implementation of a protocol through all the states and transitions defined in the specification of the protocol to determine whether the implementation conforms to its definition. And even though a protocol has been implemented correctly, it is still necessary to make sure, through *interoperability testing*, that it behaves properly when deployed as part of a complete working system.

Through the process of protocol verification, conformance testing, and interoperability testing, reliable communication products that satisfy user demands are being developed. Without reliable, efficient, and interoperable products, users cannot embrace open standards. They will be forced to depend on proprietary solutions.

#### **In This Issue**

The papers in this issue are devoted exclusively to the verification and testing of protocols. They were selected to show the importance of protocol verification and testing to AT&T and to give a small, but representative, sample of the broad range of work going on in this area within the company. They also show the beneficial impact of algorithms and formal methods on the field of protocol specification and testing.

The first paper, by Bertine, Elsner, Tewani, and Verma, reaffirms the commitment and support that AT&T gives to protocol standards in the telecommunications industry. It also presents an overview of the

methodologies and programs used within AT&T to test protocols.

The next three papers discuss techniques for testing and assuring the logical correctness of a protocol specification. The paper by Uyar, Lapone, and Sabnani shows how a protocol can be represented as a collection of communicating finite-state machines that can be combined into single global machine. A procedure is given for minimizing the state explosion commonly encountered when the component machines are combined, along with a method for analyzing the logical behavior of the global machine. The paper by Holzmann presents an efficient algorithm that can be used to search finite-automaton state spaces containing millions of states in a few minutes of computer time. The paper by Har'El and Kurshan describes a software system, based on a mathematical theory of regular expressions, that has been used to implement communication protocols by following a formal top-down development procedure that preserves the logical consistency of the original protocol specification. The fifth paper, by Sherif and Uyar, gives a generic method for modeling a protocol so that an algorithm can automatically generate test sequences for it.

The remaining papers discuss various aspects of conformance testing. The paper by Bush, Rasmussen, and Wong describes in some detail the procedures used to perform conformance tests on Open Systems Interconnection (OSI) protocols. The paper by Dahbura, Sabnani, and Uyar presents an effective procedure for generating an optimum set of conformance tests. The tests are obtained by using a finite state machine representation of a protocol to formulate an efficient algorithmic technique for finding minimum-cost tours in a directed graph.

The ability to realize a conformance test methodology depends on the testing interfaces available and the ability to coordinate test stimuli through those interfaces. The final paper by Hubbard describes a solution for the synchronization of multiple testing interfaces through the insertion of control statements in automated

---

test programs.

Even in the few papers in this special issue, we see considerable evidence of the remarkable breadth and depth of the subject of protocol testing and verification.

Biographies (continued)

*company in 1982 and supervises the Applications Architecture Planning Group at Holmdel. He is responsible for applications architecture planning. He has a B.S.E. in electrical engineering from Princeton University and a M.Sc. in Interdisciplinary Studies from the University of British Columbia in Vancouver.*

*(Manuscript received July 5, 1989)*

---