IIASA'S GATEWAY SYSTEM AND EXPERIMENTS IN DAILY OPERATION

A. Labadi

March 1981
WP-81-37

INTERNATIONAL INSTITUTE FOR APPLIED SYSTEMS ANALYSIS
A-2361 Laxenburg, Austria

ABSTRACT

At IIASA a Hungarian-made minicomputer, the TPA 70, has been put into operation to serve as a switching node for some of IIASA's remote computer connections. The system is designed to handle asynchronous data communications lines of different speeds connected to computers and terminals. It provides simultaneous access and reliable data transmission between host-terminal pairs, and also permits a kind of remote user support. For security reasons, the overall control of the connections can be carried out either by the system automatically, or by the operator of the node.

CONTENTS

IIASA'S GATEWAY SYSTEM AND EXPERIMENTS IN DAILY OPERATION

A. Labadi

INTRODUCTION

At IIASA a number of leased lines have been put into operation to provide IIASA users with on-line access to remote computing resources, databases and networks, and also to enable remote users from collaborating institutions to access resources available at or through IIASA. Most of the lines carry interactive traffic to databases and time-sharing computers.

A Hungarian-made minicomputer, the TPA 70, is used to handle most of the interactive lines, in order to provide a rather general switching facility for both inhouse and remote users.

The present hardware configuration of the TPA 70 is the following:

- 48 kbyte of CPU core memory
- 5 Mbytes disc capacity
- Teletype
- Console display
- Line printer
- Card reader
- Paper tape reader
- 3 synchronous line interfaces
- 7 asynchronous line interfaces

The following host lines are presently connected to the TPA 70:

-- PDP 11/70 inhouse computer at IIASA	1200 baud
-- TYMNET/TELENET network	1200 baud
-- IAEA database center in Vienna	300 baud

--	ESA database center in Frascati	300 baud
--	IBM 370/158 time-sharing computer in Pisa	300 baud

Terminal lines:

--	Local terminal 1	9600 baud
--	Local terminal 2	1200 baud
--	Terminal in Budapest	300 baud

The configuration can be changed easily, and in a flexible way, according to actual needs.

The aim of this paper is to give an overview of the so-called IIASA Node based on the TPA 70, to describe its capacity and capabilities and to summarize the experience gained over a 5-month experimental operation period in an international environment.

MAIN CHARACTERISTICS OF THE TPA 70 IIASA NODE

The present version of the IIASA NODE can only handle asynchronous lines. It is intended to work as a switching center for interactive traffic. Terminals and hosts using the ASCII code set can be connected to the node with a line speed varying from 50 to 9600 baud. The node system is a fully automatic one, but the operator of the node has the possibility to change almost any of the operational parameters at any time, if necessary.

The main function of the node is to provide concurrent connection for host-terminal pairs. In principle, any terminal can access any of the hosts; thus the node allows a kind of dynamic allocation of the hosts connected to it. The connections can be fully transparent, but a certain amount of filtering and limited code conversion can also be performed. The necessary administration of the opening and closing of any connection is minimized in order to make the node as "invisible" to the users as possible. However, several features are included to provide a kind of remote user support, to enable fault detection and investigation, and to ensure overall, centralized control of the whole system.

One of the terminals can be described as a monitoring device which can display the traffic of any other terminal, having a visual copy of the terminal traffic to be monitored. With the help of terminal-to-terminal messages, instructions can be given to users in trouble using a particular system. The Duty Officer sitting at the monitoring device can also issue commands to the host connected (this can also be seen by the user), thereby also providing a limited kind of remote training.

The communications lines can run at different speeds, even when they are interconnected. In order to avoid loss of data, "flow controls" are applied.

For security reasons, a strict authorization procedure is also needed. Selectively for each terminal line, it is possible to define which hosts it is permissible to access. This can be

carried out automatically by default values, but can also be changed at any time - if necessary - by the console operator.

The node also monitors the status of the lines. In case of any major failure (e.g., "carrier off"), a warning is sent to the operator and the particular line (if any) is disconnected.

A so-called day-file is maintained by the node containing all commands issued by any of the terminals or the operator, as well as status reports relating to changes in the status of the lines. Information regarding the date and time of the connections, and the number of characters transferred in both directions, is also kept in the day-file.

OPERATION OF THE NODE

The node system knows about the operator console and about the terminal and host lines. Some of the terminal lines can be described as "super terminals", which means that most of the operator commands issued are also accepted by the node.

A terminal can work either in "command mode" or "data mode". A terminal is in command mode if it is not connected logically to any of the hosts available through the node. In command mode, all the lines issued by the terminal are intended as commands for the node. A command is given in a command line terminated by "return". A command line contains the abbreviation of the command followed by the parameters (if any). The abbreviation of the command and the parameters are terminated by "space" or "return". In command mode, the "back space" (CTRL H) can be used to edit the line and "CTRL U" can be used to delete all characters previously typed into the recent line.

A terminal is in "data mode" if it is connected logically to a host. The terminal automatically goes into data mode after having issued a successful "GET" command. It remains in data mode until it has released the connection; that is, until it has issued a "DROP" command. In data mode, the terminal is in a transparent type of connection with the host; all input of characters from either the terminal or the host will then be transferred to the destination immediately.

When connection to a host has already been established, the character "CTRL A" is used as the "escape signal", to allow the user of the terminal to enter a command for the node. The character "CTRL A" has to be typed in the first position of the line; only those characters count which have been entered by the terminal user. "CTRL A", however, only represents an escape signal when it has been entered first after a "return"; possible output characters of the host between the "return" and "CTRL A" of the user are of no relevance. The "escape" is valid for one command only; that is, the response to the command returns the terminal to data mode, except if the command is "DROP". In order to send the "CTRL A" to the host in the first position of the line, a special node command is used. Of the 17 user commands available,

the "GET" and "DROP" commands are the most important ones for opening and closing a particular connection.

The operator console works in command mode only. The format of the input is the same as that at any of the user terminals. The output is line-oriented rather than character-oriented. 28 commands are available to the operator for supervising the node, such as commands to initiate monitoring of a given user terminal; commands to set lines to off-line or on-line; those to prohibit certain accesses; those to ask for different kinds of reports; and those to send messages to terminals, etc.

Every major occurrence in the system can be reported on the operator console, including all commands issued by the terminals.

SYSTEM GENERATION

At the generation time of the node system, operational parameters can be defined selectively for each line. These parameters basically control the operation of the node; most of them can, however, be changed at run time by the operator of the node. The following can be defined for each particular line:

- speed of the line (50-9600 baud)
- sizes of the I/O buffers allocated to that line
- whether stop bit error is intended as a break signal
- whether input parity is to be checked at all, and if so, by odd or even
- which output parity is to be used (odd, even, clear, set)
- if "return delay" is needed, what is the code of the delay character and how many are to be used after the return
- if "carrier-off" is to be detected, what is the length of waiting time tolerance before the action
- if flow control exists either on the input or on the output (selectively), which are the "STOP" and "START" characters
- the default logical status of the line (on or off)
- if either input or output or echo "filtering" is needed (selectively) which filtering profile is to be used (there are 11 built-in profiles, such as all the control codes, except "return" and "line feed"; or the "NULL" and "DEL" characters; or the "CTRL Q" and the "CTRL S" characters, etc.)
- whether automatic upper case - lower case conversion is to be used
- description of the logical name of the line used in the commands
- definition of the host or terminal or super terminal lines
- if it is a host line, whether remote echo is provided by the host
- if it is a terminal line, which host lines can be accessed (authorization function)
- if it is a terminal line, what is the code of the filter character (to be used to escape from data mode to issue a node command)

-- if it is a terminal line, whether "no active" status is to be monitored, and if so, how long is the waiting time (no characters are transferred in both directions) for disconnection by the node.

STATISTICS, OPERATIONAL EXPERIENCES

The present configuration of the TPA 70 was installed at IIASA in December, 1978. In February, 1979, a two-day demonstration was organized in Budapest on the computers and network connections available through the IIASA Node. Although the earlier version of the node was much less powerful, the demonstration was still successful.

During the period from October, 1979 to May, 1980, the node was brought to its present stage of development. The regular experimental service started in July, 1980 and, since then, the node has been in daily operation.

Up to now (as of November, 1980), the only regular remote user of the node has been the Hungarian National Member Organization. Monthly statistics on the Budapest terminal are shown in Figure 1. The monthly average is 74 hours, of which 56% represents access to the PDP 11/70, and 44% represents access to databases such as CEMABS at ESA, INIS/AGRIS at IAEA, Lockheed on TYMNET/TELENET, etc.

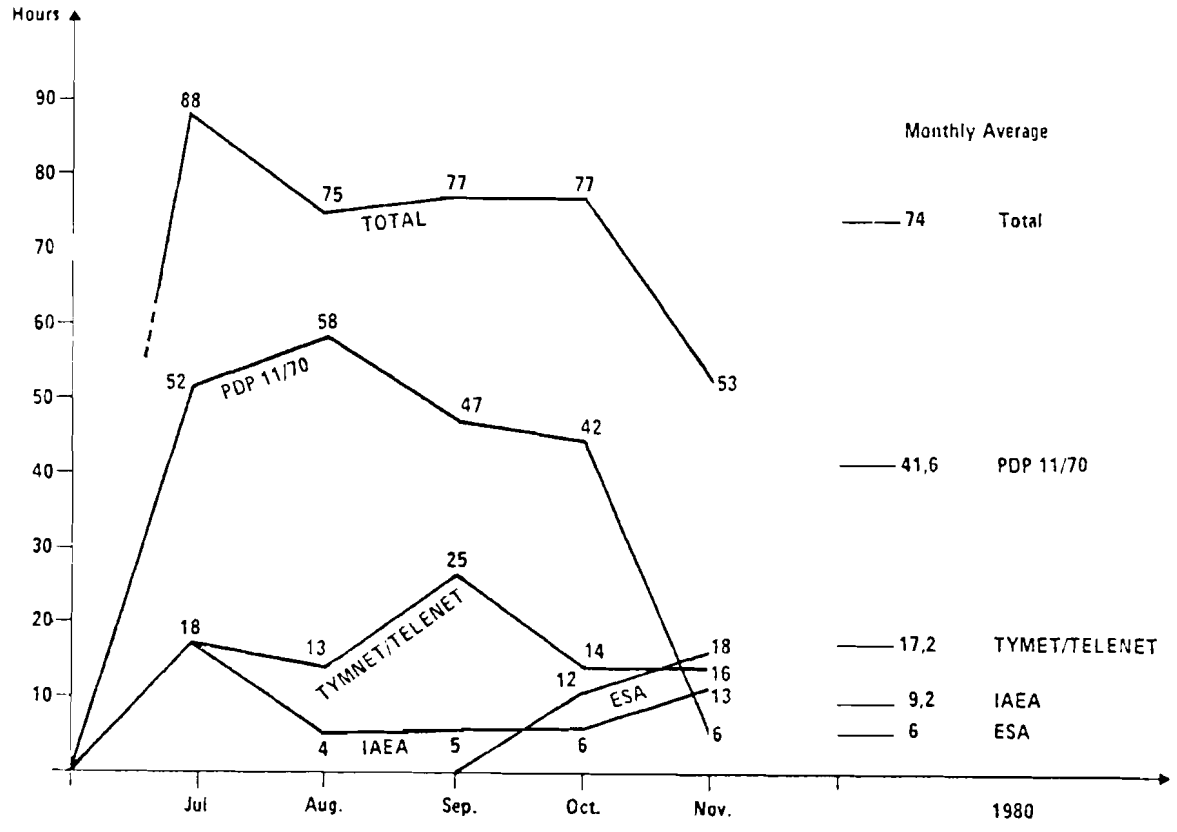

Figure 1. Traffic of the Budapest Terminal.

In June and October, 1980, a leased line was established temporarily between IIASA and the Institute for Systems Studies in Moscow, for one week each. It was a hardware-multiplexed line, and one of its channels was connected to the node. Approximately 20 hours of connection time was used to access computers available through the node.

During the 2nd IIASA Conference, the Budapest line was used to access the CDC 3300 at the Computer Center of the Hungarian Academy of Sciences, and the SIEMENS 7755 at the Institute for Coordination of Computer Techniques.

During the first period of operation of the IIASA Node, we faced difficulties many times - incompatibilities which influenced the development of the node. A few of them are mentioned below:

Computers use different strategies for the so-called "return delay"; most of them operate with either "NULL" (hexadecimal :ØØ) or "DEL" (hexadecimal :FF) characters. It happened that certain hard copy devices would not accept the "NULL" character and, for some video terminals, the "DEL" character represents a displayable code: thus, in a sense, it destroys the picture to be displayed. In order to overcome this problem, a filtering function was adopted. It can be used for input and/or output and/or echo.

As we have learned from experience, some terminals cannot generate all the possible codes of the 7-bit ASCII code set, and therefore certain functions to be initiated by one of these missing codes cannot be called. A special node command was introduced to define a character to be sent to the line by its hexadecimal value.

Certain applications use the "break" function to "kill" the process currently running. Terminals usually have a special key to provide a break in the line. Unfortunately the "stop bit error condition" (which is similar to the break) can also occur due to noise on the communication line. Within the node there is an installation parameter, whether or not the "stop bit error" condition is intended to be a break function. For this reason, a special node command had to be introduced to send a break onto the host line.

FUTURE

The capabilities of the node will be further developed: in January, 1981, 5 more asynchronous line interfaces will be added to satisfy growing demand; in March, 1981, the system will be upgraded by direct memory access boards and bit-stuffing interfaces to provide higher throughput and to create the possibility of accessing X.25-type networks.

The new version of the node (under development) will include procedures to provide access according to the recommendations of CCITT (X.25, PAD). The X.25 interface will also be used to connect a remote concentrator(s). In Budapest, a similar TPA 70 system will come into operation in the first quarter of 1981.

It will concentrate the Hungarian terminals and host(s) to be connected to the IIASA Node, and will also provide access to the Hungarian Academic Network. The Budapest TPA 70 may also receive communication lines from other countries. The two TPA 70's will be connected by a 2400 baud line using the X.25 protocol.