CYBERAIL SIMULATION

Philip G. Drew and Stephen Swerling
Arthur D. Little, Inc., Cambridge, Massachusetts

Allen B. Fonda
Castle Automated Systems, Rochester, New York

INTRODUCTION

Castle Automated Systems manufacturers Cyberail, an automated system for transporting materials within hospitals--meals, linen, trash, mail, paperwork, drugs, and so forth. Although early models of the system are in use in hospitals in Europe, and Cyberail is planned for installation in a number of new hospitals in the U.S., there is as yet no operating experience with the new system. Early in engineering development, Castle recognized that it would be quite possible to design a system, all of whose components worked satisfactorily, but which would be considered unsatisfactory by users if they found excessive delays due to poor layout or lack of equipment.

Designing a suitable configuration of tracks, switches, stations, and parking for a particular hospital was a problem which rapidly outran one's ability to analyze mentally. Therefore Castle retained Arthur D. Little, Inc., which was doing a major portion of the engineering design, to construct a simulation of the system. The intent was to have a computer model on which proposed configurations of equipment could be subjected to the anticipated pattern of demands in a particular hospital. In this way Castle could be assured that its proposed layouts would not produce excessive delays due to bottlenecks or shortages and yet would be no more elaborate and expensive than necessary.

DESCRIPTION OF THE CYBERAIL SYSTEM

Material transported by Cyberail is carried in containers about half the size of a desk, fitted with doors and movable shelves. Castors are mounted on the bottom so that a container can be moved about by hand, distributing or collecting materials locally. Pick-up and discharge of containers at stations throughout the hospital is fully automatic.

The containers are carried by individually powered trolleys, called transporters, running on a monorail within the walls of the hospital. Figure 1 gives an idea of the construction of the containers and transporters. Tracks are generally one-way and can run vertically or horizontally and in vertical and horizontal curves. At appropriate places, switches, similar to those in railroad roundhouses, provide automatically settable connections between various parts of the system. The tracks are divided by electrical interlocks into sections, each of which can be occupied by only one transporter at a time, in order to prevent transporters from bumping into one another.

In a typical sequence of operations a nurse might place a container of soiled linen, say, near the pick-up point and press a button indicating the laundry as the desired destination. If she does nothing further the station will intercept a passing transporter, but she can also press a button requesting dispatch of an empty transporter from parking. When the transporter arrives, the container is automatically moved into the shaft and connected to a yoke on the transporter, an operation taking about half a minute. The transporter stores the destination code in a set of relays and moves through the system, setting switches as it goes to the correct position. After discharging the container, the transporter is available to pick up another container or to return to parking to await another call. A schematic representation of a typical hospital layout is shown in Figure 2.

DEMAND PATTERN

The essential input to the simulation is the demand pattern to which the Cyberail System will be subjected. There have been some studies of hospital traffic flow[1] but, because the existence of Cyberail will to some degree change the patterns of flow, the question really has to be rethought for each new installation.

The procedure used at first was to make tables by hand indicating demand for a particular number of containers to be transferred from one station to another at some time within a 15 minute interval. These 15 minute groups were then used as the basis for a GENERATE block in the GPSS program. Later on, the program was changed so that a FORTRAN preprocessor punched GPSS FUNCTION follower cards to produce the desired demand pattern.

In some cases the approximate time is established by policy or by convention--for example, mealtimes are typically prescribed. In other cases hospital planners had to specify arbitrarily the approximate time of a demand--for example, within broad limits the time at which trash is emptied is immaterial and can be scheduled for a time when Cyberail is not otherwise kept busy. Thus the planning necessary for the simulation provided an operating schedule for the hospital, which has proved to be of value in itself. A number of "random" demands were also introduced to cover needs which are not as easy to specify in advance as distribution of meals or collection of linen or trash.

DESCRIPTION OF SIMULATION

Several thousand planned demands may occur in the course of a day. It proved convenient to establish demands by having an expert in hospital operations fill out tables specifying traffic between stations for different periods of the day. To reformat this information for GPSS, a FORTRAN preprocessor introduces randomness in the demand sequence, orders demands chronologically, and punches cards in the format of GPSS FUNCTION follower cards. The output of this program is a set of four decks, one each for time of demand, station initiating demand, ultimate destination, and type of demand (whether a container is needed or not).

An important design criterion for the simulation was to allow construction of the program for a new hospital configuration with minimum effort. For this reason the program was written in modules corresponding to physical portions of the system--track sections, switches, stations, parking, and so on. Each module contains GPSS blocks which assign parameters peculiar to that module, but most of the operations are accomplished in subroutines. Thus, there are subroutines for each module as well as special subroutines for demands, and for initiating required printout.

About 60 blocks are required for a station subroutine and two or three different kinds of stations exist. Six blocks suffice for a track section, and three different kinds of track sections exist. About 30 blocks are required for each switch.

Once a Cyberail layout has been prepared for a hospital, the analyst prepares a simulation deck in a straightforward fashion. All the required subroutines and the FORTRAN punched demand decks are placed at the beginning; GPSS blocks representing the system itself are then assembled with subdecks, the sequence of the subdecks corresponding to the physical system. This approach produces a program which is relatively easy to alter or to debug. It is also relatively economical in compilation time and memory requirements.

Typical output from a simulation run includes statistics for:

- time needed to fulfill demands for each station,
- delays in acquiring transporters from the parking area,
- switch usage,
- container usage,
- traffic jams,
- demands per unit time.

As with any discrete simulation, data can be accumulated on virtually any measure of system performance.

The hospitals which Cyberail has been proposed for generally result in simulation decks of about 1000 GPSS blocks, and fit without serious difficulty on an IBM 7094 with a 32K memory. Running times to simulate a 24 hour day are typically 25 to 30 minutes.

EXERCISE OF SIMULATION

We have performed simulations for three hospitals on an IBM 7094 in Boston; Castle Automated Systems is now utilizing the simulation program on an IBM 360/65 in Rochester as part of a system for each new proposed installation.

The simulation has proved to be a practical tool for exercising an operating plan and examining in detail activity within a proposed system. It has provided insight into sources of delay which were not originally anticipated. For example, in one proposed configuration the source of delays turned out to be the exit rate from parking, which was constricted by making track sections too long. It has also provided confidence that proposed configurations really are sound.

CRITIQUE OF GPSS

The original reasons for choosing GPSS were that it is a highly developed language which would be relatively easy to learn and use by people whose primary concern is hospital design and that it could be run on the computers available in Rochester. On these grounds it has proved a good choice.

In our experience, however, we have found several characteristics of GPSS III which hampered our work. These include:

- inability to insert FORTRAN statements,
- lack of flexibility in format of tables,
- no provision for inserting table headings or comments on output listings,
- inappropriate alternatives for producing transactions (entities) with the GENERATE block.

Most of these objections do not apply to GPSS 360, which is now being used. However, other difficulties in our work have come to light (aside from the temporarily unavoidable bugs in the new GPSS program, HASP, and DOS). In particular, our work would benefit from the following modifications:

- more explicit and flexible control of printout, a means of scanning all the transactions on a chain for a parameter match, unlinking the first which matches, and relinking in the same order all those which failed the match, without branching to any other part of the program.

REFERENCE

[1]James J. Souder et al., Planning for Hospitals, American Hospital Association, Chicago, 1964.

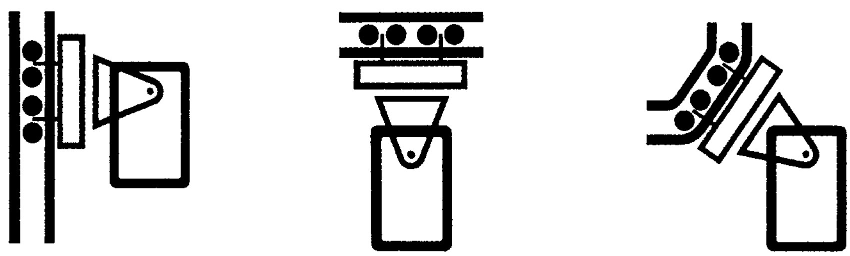

Figure 1. Sketch of Cyberail Transporter and Container. The transporters are individually powered and run horizontally, vertically, or on curves on a monorail. The container, approximately three feet wide, four feet high and 20 inches deep, is suspended from the transporter with a linkage which always keeps it upright regardless of the attitude of the transporter. The containers have doors on the front and castors on the bottom so that they can be rolled about by hand after delivery at a floor.

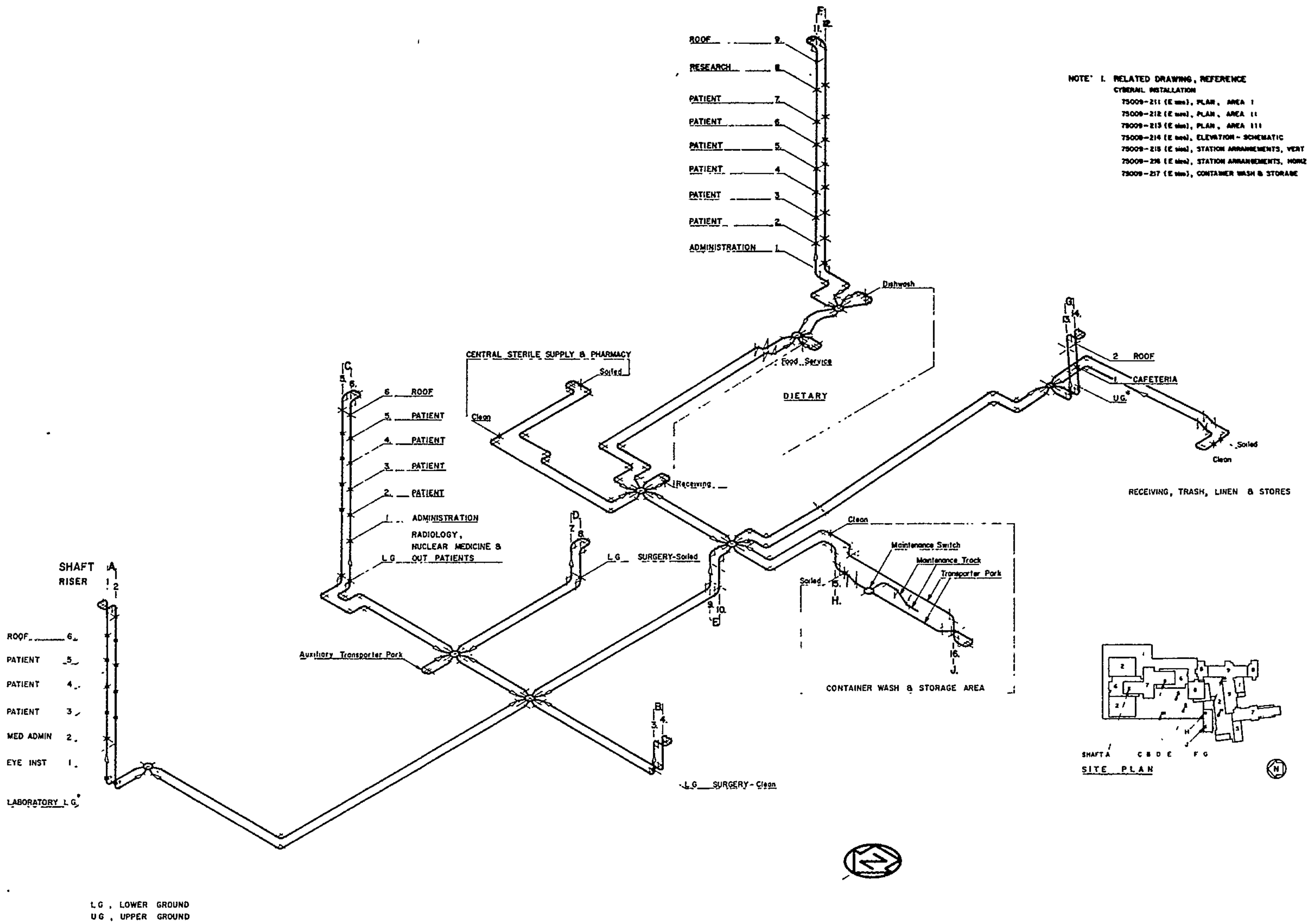

Figure 2. Sketch of Cyberail Layout in Typical Hospital