GEORGIA INSTITUTE OF TECHNOLOGY **OFFICE OF CONTRACT ADMINISTRATION**

PROJECT ADMINISTRATION DATA SHEET

☒ ORIGINAL □ REVISION NO. ______

Project No. E-24-609 (R-6040-0A0) GTRC/~~GIT~~ DATE 10 /09 /85

Project Director: D. B. Young School/~~Lab~~ ISyE

Sponsor: US Defense Supply Service
Washington, D. C. 20310

Type Agreement: P.O. # MDA903-85-M-8415

Award Period: From 09/03/85 To 01/15/86 (Performance) 03/02/86 * (Reports)

Sponsor Amount:	This Change	Total to Date
Estimated:	$	$
Funded:	$ 19,786	$ 19,786

Cost Sharing Amount: $ N/A Cost Sharing No: N/A

Title: Automated Project Management System for Z-120 microcomputer.

ADMINISTRATIVE DATA OCA Contact Ralph Grede X 4820

1) Sponsor Technical Contact:

Mr. John R. Bramblett
Defense Supply Service
Room 1D-245, The Pentagon
Washington, D. C. 20310
(703) 664-2289

2) Sponsor Admin/Contractual Matters:

Mr. Charles R. Singman
Defense Supply Service
Room 1D-245, The Pentagon
Washington, D. C. 20310
(703) 697-7766

Defense Priority Rating: S-1 Military Security Classification: Unclassified
(or) Company/Industrial Proprietary: N/A

RESTRICTIONS

See Attached Governmental Supplemental Information Sheet for Additional Requirements.

Travel: Foreign travel must have prior approval – Contact OCA in each case. Domestic travel requires sponsor approval where total will exceed greater of $500 or 125% of approved proposal budget category.

Equipment: Title vests with Sponsor; However no equipment is proposed.

COMMENTS:

* This determination from SOW paragraph 3.0 - Deliverables.

COPIES TO: SPONSOR'S I. D. NO. ______

Project Director
Research Administrative Network
Research Property Management
Accounting

Procurement/GTRI Supply Services
Research Security Services
Reports Coordinator (OCA)
Research Communications (2)

GTRC
Library
Project File
Other A. Jones

GEORGIA INSTITUTE OF TECHNOLOGY   OFFICE OF CONTRACT ADMINISTRATION

J-N-8
SR-781

SPONSORED PROJECT TERMINATION/CLOSEOUT SHEET

Date November 20, 1987

Project No. E-24-609   School/XXX ISyE

Includes Subproject No.(s) N/A

Project Director(s) D. B. Young   GTRC / XXX

Sponsor Defense Supply Service - Washington, DC 20310

Title Automated Project Management System for Z-120 microcomputer.

Effective Completion Date: 1/15/86 (Performance) ______ (Reports)

Grant/Contract Closeout Actions Remaining:

Fixed price P.O.

- ☒ None
- □ Final Invoice or Final Fiscal Report
- □ Closing Documents
- □ Final Report of Inventions
- □ Govt. Property Inventory & Related Certificate
- □ Classified Material Certificate
- □ Other ______

Continues Project No. ______   Continued by Project No. ______

COPIES TO:

Project Director
Research Administrative Network
Research Property Management
Accounting
Procurement/GTRI Supply Services
Research Security Services
Reports Coordinator (OCA) ✓
Legal Services

Library
GTRC
Research Communications (2)
Project File
Other Hutchison
Embry
DuBose

GEORGIA TECH 1885-1985

DESIGNING TOMORROW TODAY

Georgia Institute of Technology
School of Industrial and Systems Engineering
Atlanta, Georgia 30332-0205
(404) 894-2300

4 August 1987

Reference: "Automated Project Management System for Z120" (SARA)
Our project E-24-609
Your order MDA903-85-M-8415

Mr. Jesse Cox
Defense Systems Management College
PMSS Directorate
Ft. Belvoir, VA 22060

Dear Jesse:

We are pleased to enclose the Final Report for the SARA system, which includes the following items:

1. The SARA User Manual on Wordstar diskette under filename USERSARA, excluding the separate Appendix 6.5

2. The complete source code on a set of diskettes.

3. The complete executable code on a set of diskettes.

4. The Program Documentation Appendix 6.5 of the User Manual in hardcopy form.

We believe that this Final Report is complete and will give you an easy-to-use, easy-to-change scheduling software system that the Defense Systems Management College will find very useful both in teaching project management to students and in scheduling DSMC research and development projects.

We will continue to be available informally to answer questions and give advice, especially with regard to the repair of minor deficiencies. We would very much like to be given the opportunity to bid on further enhancements of SARA.

Sincerely,

Donovan Young
Associate Professor

USER MANUAL

SCHEDULE AND RESOURCE ALLOCATION

(SARA)

MODULE

A scheduling and resource planning tool
for projects
with multiple constrained resources

TABLE OF CONTENTS

1.1 Brief Description of SARA

SARA (Schedule And Resource Allocation) is an interactive, small-computer program to aid a user in scheduling a small- to medium-size project (up to 75 activities and 16 resources) where there is contention among activities for scarce, shared resources. Several of the various activities in a software development project, for example, may need to use designer labor, programmer labor, computer access, and other shared resources. If the user defines the activities, their durations or duration ranges, their precedence relationships, their resource consumptions, and the availabilities of resources, then SARA can help the user schedule the project. The aim of scheduling would normally be to find the shortest schedule among those whose activities are sufficiently spread out to keep resource consumptions within availability limits. SARA is designed to help the user develop and enter the data and to perform the scheduling interactively. The user makes trial changes in the schedule, or accepts scheduling recommendations from SARA's scheduling heuristic, and can immediately see the consequences. For example, the user may reduce the duration of a critical activity; SARA will respond by rescheduling, and the user can see, in detail, if there is any reduction in the project duration and whether there are any resource overconsumptions caused by the change, including, for example, an overconsumption caused by earlier starting of a later activity whose start time directly or indirectly responds to the reduced duration.

This user manual gives simple instructions as to how to invoke and use SARA. The menus on the interactive screens in SARA and the error messages provide instructions that are complete enough so that references to this manual should seldom be necessary after initial familiarization.

1.2.1 Work Breakdown and Activities

A project is a collection of activities such that if and when each of the activities are completed, the project is completed. The first step in scheduling and resource planning using SARA is work breakdown - defining the activities.

An activity should represent a cohesive effort. It has a definite start time and a definite duration, both of which are either to be fixed in advance or to be fixed during scheduling. An activity cannot be interrupted except (as for nights and weekends) when the entire project is interrupted. For purposes of enforcing resource constraints, an activity is considered to consume each of its resources at a steady rate. Resources cannot be substituted for one another except by altering project data.

Activities have precedences, by which a successor activity may not begin until the next period after the finish period of its latest-finishing predecessor activity.

Activities should be defined so as to be as separate and independent of each other as possible. In SARA, they directly interact with each other only through precedence relationships and by competing for constrained resources. The conceptual limitations on activities--uninterruptibility, steady consumption rates, lack of resource substitution, and strict precedences--are what distinguish critical-path-method (CPM) scheduling by SARA, from work breakdown, which is done outside of SARA. The best schedule is best only for a specific work breakdown; if part of an activity can be delayed, or can use different resources, or can use resources at a different rate, or can allow a successor to start before the rest of the activity is complete--then the activity should be redefined or broken into two or more activities. SARA performs scheduling and resource allocation for a project whose methods of accomplishment are already expressed by the work breakdown.

1.2.2 Constrained Resources

Multiple constrained resources are handled by SARA; this is SARA's distinction from other CPM software packages. SARA is an appropriate tool for planning a project in which one or more resources have limited availabilities so that the project cannot be completed as fast as if infinite resources were available.

A very simple example will show how resource constraints work. If you look at a college catalog while planning the project of earning a degree, each course is an activity. There are precedences; one course requires other courses as prerequisites. The longest prerequisite string (critical path) is no more than five or six courses long. Why, then, does it take a student at least twelve semesters to earn a degree? The reason, of course, is that the student's time is a constrained resource. If a course load of no more than fifteen credit hours were allowed, then not every course could be taken immediately after its prerequisites. Part-time work could constitute other activities, and money could constitute an additional resource. Whereas non-resource CPM methods would show an unrealistic schedule, SARA would show the delaying effects of resource constraints.

Each resource in SARA has availability cycles representing time spans such as days, weeks or months, of arbitrary constant length. If the cycles have a length of one period, the availability constraints are those for a fully "renewable" resource; if the cycle length is so long that there can only be one cycle in the entire project, the availability constraint is that for a "nonrenewable" resource. There can be a different availability in each cycle; for example, if a cycle represented a 21-shift week for a resource coded "PRG" and named "progrmrs", the user could specify how many programmers could be used during each week separately.

Different limitations on what is, in the real world, the same resource can be expressed as separate SARA resources. For example, another resource with a code different from PRG could be used to express a shift-by-shift limit on programmers. In one project there were 35 actual programmers available, but no one could work more than 11 shifts per week. Two SARA resources were defined: one gave a limit of 35 programmer-shifts per one-shift cycle, and the other gave a limit of 385 programmer-shifts per 21-shift cycle.

Resources are consumed by activities, and in any cycle of N contiguous periods the total consumption of a particular resource is the sum of the consumptions by each of the activities that consume that particular resource and are ongoing during all or part of the cycle. This summed consumption is what SARA compares with availability to determine the excess consumption (resource conflict) of each resource for each cycle. If a schedule has no resource conflicts, it is said to be resource feasible.

Availability segments are data that specify the amount of resource available per cycle. Each segment is a number pair: an integer giving the serial number of the first cycle in which the availability is a certain amount, and a real number giving the amount. For example, if periods represent shifts and the cycle length for PRG is one period, then the availability segments

1	35.
22	0.
43	35.

would specify that 35 programmer-shifts (i.e. man-days of programmer effort) are available per shift for the first 21 shifts, then none for the next 21 shifts, then 35 again for the 43rd and subsequent shifts. As another example, if the cycle length for "prg" (a resource code distinct from "PRG") is 21 shifts, the availability segment

1	385.

would specify that, beginning with the first 21-shift cycle, only 385 programmer-shifts are available per 21-shift cycle. The effect of defining SARA resources "PRG" and "prg" and giving them these availabilities is that SARA would show resource infeasibility for "PRG" if a schedule proposed to use more than 35 programmers at any one time (or any programmer in the shifts numbered 22 through 42), and would show resource infeasibility for "prg" if a schedule proposed to use more than 385 programmer-shifts of effort in any one of the defined 21-shift cycles.

1.2.3 Consumption and Duration

The consumption of each resource by each activity depends on the activity's duration. Each activity has three key durations called MINimum, NOMinal and MAXimum duration. They can all be equal, or NOM can equal either MIN or MAX, or they can be three distinct durations; of course MAX≥NOM≥MIN.

For each of an activity's key durations, there is a corresponding key consumption of each resource that the activity consumes. Key durations and consumptions are provided as data. During scheduling, when any other duration between MIN and MAX is tried, SARA interpolates consumptions according to a hyperbolic interpolation function. In this way the data requirements are held to a reasonable volume yet it is possible to consider all feasible durations instead of just three.

The hyperbolic interpolation function, which is explained in more detail in Section 6.3, is based on a semiempirical model which assumes three components of resource consumption: an intrinsic work content independent of duration; an extra content proportioned to the rate of resource usage; and an extra content proportioned to the duration. The interpolation formula is

where x is the duration, y is the consumption (total consumption of the resource by the activity), and the constants C_1, C_2 and C_3 are computed so that the curve passes through the three key duration/consumption points. The activity is assumed to consume Z=y/x units per period within its duration.

As an example of hyperbolic interpolation, consider an activity that can be accomplished by workers as follows:

	duration, x	consumption, Y	(consumption rate, z=y/x)
(MIN)	17 hr	170 worker-hr	(10 workers)
(NOM)	30 hr	150 worker-hr	(5 workers)
(MAX)	100 hr	200 worker-hr	(2 workers)

For this example, assuming periods represent hours, the user would input 17, 30 and 100 as the key durations and 170, 150 and 200 as the corresponding key consumptions. The computer would calculate the constants. Then, if the user were to try a duration of, for example, 50 hours, the interpolation routine would compute a corresponding consumption of 157.78 worker-hr (3.1556 workers). The sketch below shows the interpolation result to approximate scale.

1.2.4 Desired Start and Finish Times

SARA measures time internally in working periods, which most commonly represent working days. The earliest representable time in SARA is the desired start time (DS), which provides a left (earliest) reference point for time scaling. If the user does not fix the start times of activities, all activities that have no predecessor and that are not serialized (explained in paragraph 1.2.5 and 1.2.6) to follow other activities will start at DS, which is known internally as period 1 and externally either as a date or by a period number, according to the calendar option chosen.

Fixes are relative to DS. That is, if an activity has a fixed start date of, say, 15 Aug 1988 (represented in SARA as 15AU88), then if DS is changed the fixed start date will also be automatically changed by the equivalent amount.

DF, the desired finished time, is a due date for the project. It, too, is relative to DS. Time conflict is defined to occur to the extent that the project misses DF and/or violates precedence requirements and/or serializations. A time-feasible schedule has no time conflicts.

1.2.5 Precedences and Serialization

A precedence, specified in input data, and a serialization (quasi-precedence), specified in scheduling, are both pairwise relationships which provide for a predecessor activity to finish before a successor activity can begin. Specifically, precedences and serializations provide that (1) automated scheduling will not schedule an activity to start before all its predecessors or quasi-predecessors finish, (2) the heuristic will not contemplate scheduling an activity to start before all its predecessors or quasi-predecessors finish, and (3) if the user fixes an activity's start period on or before the finish period of a predecessor or quasi-predecessor the schedule will be reported as time-infeasible, and negative slack indicators on the Gantt chart will show how far the precedence-violated activities would need to be shifted left or right to obey the precedences or quasi-precedences.

SARA treats the due date (DF) as if it were a fixed successor to the latest-finishing activities. Negative slack and time infeasibility from failure to meet the due date are reported; if a schedule violates precedences or quasi-precedences and also fails to meet the due date, the reported time-infeasibility is summed, but the negative slack indicators do not extend beyond the needed distances to obey precedences, quasi-precedences and the due date.

A precedence is intended to represent a technological necessity to complete one activity before starting another (for example a roof cannot be installed before its supporting walls).

A serialization (quasi-precedence), on the other hand, is intended to represent the user's intention to schedule one activity after another one (presumably for the purpose of avoiding resource conflicts). Precedences are part of batch data input and apply to all plans for a project; serializations apply only to the plan in which they are specified. (See paragraphs 1.2.6 and 4.1 for a definition of a plan.)

1.2.6 Scheduling

Scheduling in SARA is done interactively with the help of the Gantt Chart Screen, which displays a bar-chart schedule, resource conflict graphs, and a resource consumption/availability profile, all on the same time scale. The user views the schedule, makes changes, and views the schedule again to see the consequences.

There are three kinds of scheduling activities available to the user: serialization, start fixing, and duration changing. In serialization the user specifies quasi-precedences to force quasi-successor activities to wait for completion of quasi-predecessor activities. In start fixing the user specifies start times for activities. In duration changing the user alters the activity durations.

SARA computes the consequences and displays them. For example, when the user changes the duration of an activity, SARA computes the new timespan of the activity (from its start time to a new earlier or later finish time), computes the activity's new resource consumptions (see paragraph 6.3, Hyperbolic Interpolation), computes the new resource consumption rates over the new timespan, reschedules all the activities that have the activity as a direct or indirect predecessor or quasi-predecessor, recomputes all the resource consumption profiles, compares them with availabilities, and computes resource conflicts.

A saved schedule in SARA is called a plan. Serialization, start fixing and duration scheduling apply only to one plan. The various plans for a project represent different schedules for the same project. Up to five plans can be saved for a project. (See Chapter 4 for scheduling procedures.)

From the point of view of the user's purposes in performing scheduling in the multiple-constrained-resource environment for which SARA is designed, most scheduling activities may be considered as ways of converting resource constraints to time constraints so that they can be enforced by SARA's scheduling computations. For example, a string of activities can be serialized to keep them from consuming the same resources at the same time; or a start time may be fixed at a time that is later than the activity's "early start time" so that it will use resources at the exact time they are plentiful; or a duration could be changed to conserve resources or to exploit a high availability over a short timespan.

Even before the user does any scheduling, there is one plan defined by default and labeled as Plan 1. This is the early-start plan that obeys all precedences but ignores resource availabilities.

In SARA scheduling, it can be said that SARA takes care of time while the user takes care of resources. That is, SARA merely "complains" about resource conflicts but ignores them in computing the schedule, which is computed on the basis of starting each activity as early as possible while obeying all precedences and quasi-precedences, obeying all fixed start times whether from the data or from a plan-specific user adjusted start time, and using the scheduled durations specified in data or by plan-specific user adjusted durations. Because SARA has no control over fixed start times and the project due date (DF), it cannot totally enforce time feasibility; it "complains" about any time conflicts forced upon it.

Time conflicts are displayed in the form of negative slack lines (solid lines) on the Gantt Chart, which extent to the left or right of activity bars to indicate how far the activity's start or finish time would need to be altered to achieve time feasibility. Thus there are, for example, negative slack lines to the left of critical activities when the schedule extends beyond the project due date (DF). (Positive slack lines (dotted lines) are also shown to indicate how far an activity could be delayed without causing time infeasibility.)

1.3 Hardware ans Software Limits

Hardware: SARA requires a Z100 military standard microcomputer with a hard disk, 640K of memory, and screen graphics.

Software: A plan with up to seventy-five activities and sixteen resources can be scheduled on SARA. There can be a maximum total of 150 precedences and 150 serializations. The specified schedule cannot last for over 104 time periods (two calendar years if the periods are defined as weeks). There is plenty of room for resource availabilities and code weight pairs, so the user should never encounter any problems with these constraints. File names can be up to eight alphanumeric characters (the first character must be a letter) followed by a period and an up to three character alphanumeric extension. The schedule screen can display up to 22 activities at a time, or a total of 22 activities and resources, or 18 activities and a resource profile.

1.4 Time Scaling in SARA

Time is represented in SARA in terms of working periods. A working period, or period, is an indivisible measure that can represent any working time span (such as, for example, five working days, which could be, say, Thursday and Friday of one week and - if the following Monday were a holiday - Tuesday through Thursday of the next week). Periods are numbered serially, and SARA can work with up to 104 of them, such as periods 1 to 104, or 208 to 311. The timescale labels for periods are of two alternative forms depending on the calendar options described in 1.4.1, and there are two special periods - desired start (DS) and desired finish (DF) - that fix the earliest representable time and the project due date, respectively, and are explained in 1.2.4.

1.4.1 Time Boundaries

The earliest and latest times shown on the Gantt chart screen are time boundaries, which are set by the user on the schedule screen using the "Set time boundaries" menu item. The user enters the appropriate code, depending on whether the NORMAL or WORKDAY calendar option is in effect (see paragraph 1.4.2).

The Gantt chart can show up to the full 104 time periods that SARA recognizes, but for the user to see small differences, it is good practice to show no more than about 52 periods at once. The default time boundaries are 1 and 52. The Gantt chart is easiest to work with when the boundaries span about 35 periods. (With 104 periods the time ticks are three pixels apart; with 52 periods, they are six pixels apart; with 39 periods, they are eight pixels apart, and slightly over a third of the maximum time span is viewed at one time.)

1.4.2 Calendar Timescale Options

There are two calendar options for labeling of SARA timescales : NORMAL and WORKDAY. The NORMAL option is set up for simple serial numbering of periods; it is especially appropriate when a period represents a week, and is marginally appropriate when a period represents an arbitrary working timespan, such as a shift, that is not a number of days or is not to be tied to calendar dates. The numbering system is

(01:01), (01:02),..., (01:52), (02:01),...(02:52).

The numbers XX:YY can be interpreted as "year" XX and "week" YY. In the current version of SARA the user cannot modify the 52-period modules.

When the NORMAL option is in effect, the user enters data as integers. For example, 53 would indicate the period labeled 02:01, and 70 would indicate the period 02:18.

The WORKDAY option is set up for date labeling; it is especially appropriate when a period represents one working day, and is also appropriate when a period represents N consecutive working days. The calendar date attached to a period by SARA is the first working day in the period; when the user inputs a date, the period SARA interprets as meant is the one whose first working day is nearest the user-specified date. In the current version of SARA the working days in the years are weekdays, assumed to be all Mondays through Fridays, except for the following holidays:

New Year's Day - January 1 or nearest weekday
MLK Day - third Monday in January
President's Day - third Monday in February
Memorial Day - last Monday in May
Independence Day - July 4 or nearest weekday
Labor Day - first Monday in September
Columbus Day - second Monday in October
Veteran's Day - November 11 or nearest weekday
Thanksgiving - fourth Thursday in November
Christmas - December 5 or nearest weekday

The format of calendar date codes is XYYZZ or XXYYZZ, where X or XX is the day of the month from 1 to 31, YY is a month code, and ZZ is the last two digits of the year, such as 88 for 1988. The month codes are:

January	JA
February	FE
March	MR
April	AP
May	MY
June	JN
July	JL
August	AU
September	SE
October	OC
November	NO
December	DE

The rule for remembering month codes is that they are the first two characters of the full name, except that the second character is replaced by the third if the second would not give a unique code. Thus March and May are MR and MY, and June and July are JN and Jl; January is JA, since no other month starts with JA.

2.1 Programs and Documentation

The SARA system is comprised of ten seperate programs: one for the entry screen (ENTRY.EXE), one for batch data entry (E1.EXE), two for scheduling (SCHEDULE.EXE, SCHSCR.EXE), and five for report generation (K1.EXE, L2.EXE, S1.EXE, Q1.EXE, M1.EXE). There are six reserved file names in addition to these executables. They are: SARA.BAT, the batch file that executes SARA; AAA.AAA, an original form of TFER.BAT, the actual batch file which transfers control to the various programs; GITPASE.DAT, a temporary data file; GITPASE.WRK, a second temporary data file; and GITPASE.RPT, a temporary report file. The only other necessary program is a text editor.

Appendix 6.5 contains all the program documentation.

2.2 Installation

To install SARA on a Z100 Microcomputer, all that is required is to cooy the three program disks onto a hard disk. SARA as written will not run on a merely floppy disk system.

2.3 User Entry and Exit

To start a SARA session, type "SARA" after moving to the directory containing all the SARA-executable files. To exit, select "End program" on the entry screen, or turn off the computer.

2.4 General Interaction Principles

There are a few general user principles found throughout the SARA program. The up, down, left and right arrow keys permit the user to select an action which a carriage return then activates. In any instance where a typed input is required, a null entry carriage return only will abort or skip the action. The only exception to this is the entry of a file name when saving a file. In this case, a "Q" is required to abort the save, while a blank carriage return saves to the default file name.

2.5 Control of Files and Reports

There is no special mental effort required to track files and reports. The following are either suggested or required guidelines:

1) All batch text files should end in ".TXT" (suggested).

2) All data files should end in ".DAT" (suggested).

3) All report files are the active file name with the extension ".RPT" (required).

4) Any requested report is appended to the existing ".RPT" file (required).

5) A binary (SARA-usable) file may be saved over one of the <u>same name</u> only if that name is the default file name listed in the batch (human-readable) file from which the binary file was created (required).

3. Batch Text Data for SARA

All SARA data are provided originally by creation of a batch text data file. This file contains, general information (a default file name, a project description, calendar timescale options), resource information (resource codes, names, descriptions and availability constraints), activity information (activity numbers, names, descriptions, duration/consumption relations, and scheduled durations if provided as data), activity fixed start times, if provided as data, and precedences.

SARA uses a three step process to enter data into the system. First, the text file must be created by the user in the format shown in paragraph 3.1. This text file must not contain any special ASCII control codes such as those inserted by most word processors. The text file should be created using the line editor EDLIN that comes with DOS, or a word processor mode that creates straight ASCII text, such as Wordstar's non-document mode.

The second step is to convert the text data file into binary form that is readable by SARA. This is accomplished by selecting "Batch data mode" from SARA's main entry screen. When in the batch data mode, the text file can be loaded into memory via option 1 and saved back to disk in binary form via option 3.

The third step is to Open the Binary File via option 4 on the Main Entry Screen. After this is done, your data is in SARA and ready to use.

The Batch Data Mode is a separate facility for dealing with batch data without causing the user to have to exit and re-enter SARA. The Batch Data Mode is controlled under SARA, but does not share data with it. One of the consequences of the separated structure is that the user can work in the Batch Data Mode with both batch and binary files, yet go back to SARA proper without having changed the active binary file; another consequence, conversely, is that saving a binary file in the batch screen does not make it SARA's active binary file, so the user must perform the Open a Binary File option on the Main Entry Screen even if the binary data has just been prepared in the Batch Data Mode.

The Batch Data Mode provides a computer-supported way of revising a batch text data file to reflect the potentially confusing and complex changes that occur in <u>deleting activities or resources</u>. All other project data changes are made by the user directly by editing of the batch text data file.

3.1 Batch Text Data Format

A batch text data file contains a variable number of lines of text in five areas (groups of lines) arranged as follows:

General area of batch text file:

G1 Default file name

G2 Project description, first line

G3 Project description, second line

G4 Calendar choice

If calendar choice is workday:

G5 Workdays per period

G6 Desired start date

Resource area of batch text file:

R1 Resource code and name

R2 Resource description

R3 Period in which first available cycle begins

R4 Cycle length in periods

R5 Availability segment

Further lines R5 for additional availability segments

Further sets R1-R5 for additional resources

END OF RESOURCES line

Activity area of batch text file:

A1 Activity number and name

A2 Activity description

A3 Key durations

A4 Key consumptions of a resource

Further lines A4 for additional resources

Further lines A1-A4 for additional activities

END OF ACTIVITIES line

Indirect-resource area of batch text file:

I1 Indirect consumption line

Further lines I1 for further indirect consumptions

END OF CODE WEIGHTS line

Fix area of batch text file:

F1 Activity start fix

Further lines F1 for start fixes of additional activities

END OF FIXES line

Precedences area of batch text file:

P1 Predecessor-successor pair

Further lines P1 for additional precedences

The format and content for each kind of line are as follows (all lines begin in column 1 and contain no extra blanks):

NOTE: Line labels (A1, G2, etc.) do NOT appear in the file.

Line G1 is a DOS file name (up to 8 characters with a 3-character extension) for the project. The extension must be DAT. Example: PMSSPROJ.DAT

Lines G2 and G3 are two 60-character lines that serve to describe the project.

Line G4 specifies the calendar choice. If it consists of the word WORKDAY in uppercase letters, the calendar assumes a 5-day work week with Saturdays, Sundays and standard holidays omitted; the next line (G5) is the integer number of workdays in each SARA period; and the next line (G6) is the calendar date of the DS (desired start time) for the project. If line G4 consists of the word NORMAL in uppercase letters, the periods are simply numbered serially, and there are no G5 or G6 lines. See Section 1.4 for calendar and time scale details.

Line R1 declares a resource code and name. The first 3 characters are the user-declared resource code that will be used on the screen to identify the resource; the code, of course, must be different for each SARA resource. The fourth character is blank. The resource name is the next 10 characters. When the name is printed the code is also printed, so most users use a naming style that lets the code be the first letters of the perceived name, as in the example PRO GRAMMER001. TES T LAB1 and tes T LAB2 are legal resource codes and names, and because of the distinction between uppercase and lowercase letters, they would be distinct even without the 1 and 2 in the names.

Line R2 gives a 48-character desciption of the resource.

R3 gives the integer period number or the calendar date in which the first availability cycle begins. Availability is zero before this. Examples: 2 for the second period or 4MY87 for the workday nearest May 4,1987.

Line R4 gives the integer availability cycle length in periods. Example: 3 for availability in 3-period cycles.

Line R5 gives an availability segment consisting of an integer and a real number seperated by one blank. The integer gives a cycle number (example: 2 for starting in the second cycle). The real number gives the maximum number of units of resource allowed to be consumed per cycle from the given cycle onward. Any number of lines R5 can be included to collectively define the stepwise availability function. Example: two lines, 1 30. and 5 15.5, indicate that the availability of the resource is thirty in the first four cycles and 15.5 in subsequent cycles.

Lines R1 through R5 are given for each resource.

After the last R5 line for the last resource, there is an END OF RESOURCES line containing the exact sixteen characters"END

OF RESOURCES" in uppercase letters with two internal blanks.

Line A1 gives a serial activity number, a colon, and a 12-character name. Activities must be numbered serially from 1, so the first instance of line A1 must start with "1:", the second with "2:", etc.

Line A2 gives a 48-character description of the activity.

Line A3 gives the three key durations (MIN, NOM, and MAX) as integers (seperated by single blanks), and, optionally, a fixed duration for the activity. The default scheduled duration is NOM. For example, the enrty 2 3 10 7 sets the minimum duration of the activity at 2 periods, the nominal at 3, the maximum at 10, and the scheduled duration at 7.

Lines A4, one for each resource that the activity consumes, gives the key consumptions of the resource as three real numbers. For example, the entry 10. 8. 7. specifies that the activity consumes 10 units if done in its minimum duration, 8 units for NOM duration, or 7 units for MAX. For other durations, SARA will interpolate consumptions whenever the duration is changed.

Further sets of lines A1 - A4 are entered for further activities. The END OF ACTIVITIES line (all uppercase, with single blanks seperating words) signifies the end of the activity area of the file.

Line I1 gives the code of an indirectly-consumed resource, a blank, the code of one of its consumers, and its consumption factor. The consumption factor is a positive number of up to 7 digits, expressed (with a decimal point) in up to 8 characters. For example, if RE1 represents wages, RE2 represents workers paid at rate 13.75 per period, and RE3 represents workers paid at rate 27.15 per period, then the lines <RE1 RE2 13.75> and <RE1 RE3 27.15> would express this. RE2 and RE3 could themselves be indirect resources.

After the last I1 line, there is an END OF CODE WEIGHTS line containing the exact nineteen characters "END OF CODE WEIGHTS" in uppercase letters with three internal blanks.

Line F1, one for each activity whose start time is to be fixed in the data, contain the activity serial number (an integer), a blank, and the date or period number in which the activity is to start.

The END OF FIXES line (all uppercase, with single blanks between words) signifies the end of the fix area of the file.

Lines P1, one for each predecessor-successor relationship between activities, each consist of two integers seperated by a single blank; the first integer is the activity number of the predecessor, and the second integer is the activity number of the successor.

3.2 How to Create and Revise Batch Text Data

Under MS-DOS the user can create a text file using any available editor or word processor in a mode that creates a straight ASCII text file without any special control codes (e.g., Wordstar in non-document mode). For example, if the DOS EDLIN editor is used, the user gets into the SARA-accessible directory and enters

```
EDLIN filename.TXT
```

where "filename" is replaced by the filename under which the user desires this project batch text data to reside.

Then the user creates the batch data file according to the format given in section 3.1. The first line of the file should be

```
filename.DAT
```

to provide a default name for the <u>binary</u> file that will be prepared later (although the user will be able to change that name when saving the binary file).

When the text file appears to be correct, the user exits the editor in such a way that the batch file is saved under the filename.TXT name. With EDLIN, the "E" command does this (see Chapter 12 of the Zenith Z-120 manual for EDLIN documentation).

The next step is to invoke SARA, go to the "Batch data mode," select the "Load batch data" option to retrieve the batch text data file from disk and convert it to computer memory form, then select the "Save binary data" option to convert the data from computer memory to SARA-usable binary form. If error messages are received in response to the save attempt, the next step is to go back to MS-DOS (select "Return to entry," then on the entry screen select "End session"); invoke the editor again, and change the batch text file to conform to the required format.

If no errors are noted upon a "Save binary data" in the batch data mode, there now exists a binary data file for the project under the filename.DAT name in the SARA-accessible directory.

To revise a project's data, the user can simply edit its batch text data file. However, it is necessary to exercise control over file names to avoid confusion (see paragraph 3.6.2). While there is little confusion if the new version is given the same name, it is necessary to exercise tight control. To prevent inadvertant overwriting of valid data,SARA prevents a binary file save to an existing file already in the directory, unless the name is the filename.DAT name that appears in the first line of the batch data file. It is thus assumed that the user intends a save to the default name to represent saving a replacement version of the binary file, while a save to a new name would represent a new version that would not have the same name as any other file except by mistake.

If a project's data is to be changed by deleting activities and/or resources, the user would be faced with unsavory tasks such as changing the activity numbers, reflecting the changes in lists of fixes and precedences, and removing multiple mentions and cosequences of the deleted activities or resources. Since that would be essentially as difficult as building a new batch data file from scratch, and since the consequences of deletions are well defined, SARA provides a "Delete activities" function and a "Delete resources" function in the batch data mode (see section 3.4).

In general, a revision to a batch text data file requires that the binary file saved from the old text file be destroyed (deleted) and that a new binary file be saved. Any plans (see section 4.1) under the old file become meaningless, and are destroyed when the binary file is destroyed.

3.3 How to Load and Save Batch Text Data

To load batch text data means to retrieve a batch text data file and convert it to computer memory form. This is done in the batch data mode. It is done to allow saving of the data in a binary file or to allow automated deletion of activities and resources as an alternative to tedious manual editing.

Load ing batch text data files does not interfere with Sara's active binary file, so a user could interrupt a session to work with the batch data of another project (though this would seldom be desired) without needing to reopen the active binary file upon returning to SARA proper from the batch screen.

To load a batch text data file, select "Load batch data" on the batch screen. Then provide the name of a batch file that is in the SARA-accessible directory. (The user must remember the name or must exit SARA and go to MS-DOS to query the directory.) By convention, all batch text data files for SARA contain the extension ".TXT".

To save batch text data means to convert the current batch text data from computer memory form to text form and create a batch text data file from it under a user-provided name. There are two situations in which the user would need to do this. First, the user could have deleted activities or resources using the automated methods available in the batch data screen (see paragraph 3.4), so that the file would archive these changes. Second, the user could desire, for any of several reasons, to create a batch text copy of the current data under a different name from that from which it was loaded. (Renaming can also be done in MS-DOS.)

3.4 Deleting Activities and Resources

To delete activities or resources, go to the batch screen and load a batch data file by selecting "Load batch data" and typing one of the available file names with the ".TXT" extension. SARA provides a special procedure to automatically revise a batch text file to incorporate all the consequences of deleting activities or resources. If a text file is to be prepared for a project identical to another one except for having fewer resources or activities, the text file for the "reduced" project would be difficult to prepare by editing of the old file. For example, if one resource is deleted, all references to that resource by any activities that consume it must also be deleted. As another example, if an activity is removed, all activity numbers below it and all precedence relationships must be altered to reflect the new numbering. Therefore two utilities are provided to enable preparation of a new batch file by simply specifying which activities or resources are to be removed.

To delete activities, make sure that the proper batch text file is loaded and select "Delete activities" on the batch screen. A list of activities is displayed. Select the activity to be deleted. To confirm the deletion, select the activity again.

Since the batch mode is a separate mode that is not really part of SARA proper, the results must be saved in a file to be used. To make the new data usable, select "Save binary data" on the batch screen and provide either a new name, in order to save the new data without destroying the previous binary file, or the old name, to replace the old binary file with the new. To write the new data over the old file, make a null entry in response to the "Save as which binary file?" prompt. Subsequent loading of the new file in SARA will make the new data effective.

If it desired to update the batch data file, a similar procedure can be initiated by selecting "Save batch data" on the batch screen.

To delete resources, select "Delete resources" on the batch screen and proceed as described above for deleting activities.

3.5 Binary Data Files

To do any scheduling in SARA, there must be an open binary file. A binary file is a file which, by convention, has the extension ".DAT" and is the data file that SARA uses. It contains all the plan-specific information for up to five plans (schedules) for a project. By contrast, the batch test file has the ".TXT" extension and contains only the non-plan-specific data for the project in human readable ASCII form, as discussed in section 3.1.

To create a binary file for the first time, after a batch data file has been loaded, and convert computer memory batch data to SARA-readable data, the user must select "Save binary data" in batch mode. In answer to the "Save as which binary file?" prompt, a null entry will cause an attempt to save a binary data file whose name is the default file name listed in the first line of the batch file. This naming can be overridden by supplying a new name, which should include a ".DAT" extension. An entry of the letter "Q" will abort the conversion and saving of binary data. If the conversion of batch data to binary data succeeds, the name of the binary file will be shown as the "Active binary data file" on the screen, and SARA will be ready to schedule using this data. If the conversion attempt fails, an error message will be recieved, and it will be necessary to revise the batch data file. The special diagnostic error message that is provided gives the line number at which the error occured in the batch file.

When a new plan is created and saved in the scheduling mode of SARA, it is saved only in the computer memory so that the same plan can be recovered later in the same session. For a plan to be archived so that it can be recovered in a later session or after a power failure, select "Save current binary file" on the entry screen.

3.6 Data and Name Control

SARA does not enforce the convention that batch files have the ".TXT" extension, nor the convention that the batch file and binary file for a given project have the same name, before the extension. These practices are recommended, however.

The first line of a batch text file is the reserved file name for the corresponding binary file. When a binary file is saved, by selecting "Save binary data" in batch mode, the default name is that reserved name, and the binary data will be written over previously saved binary data with the same name, if any; that is, a binary file having the reserved name specified in the batch text file can be updated. SARA prevents overwriting of files in any other manner. An attempt to save binary data to a file name that already exists, unless it is that reserved name, causes a "blocking error" and expels the user from SARA.

3.6.1 New Project Data and Name Control

Upon prparing the text file for a new project, the user should decide whether there will be more than one version of the basic project data, and whether human-readable batch text files will be maintained for each version. Since file names are limited to eight characters plus a three character extension, it is difficult to create a series of related names. The user may wish to use extensions such as ".TX1", ".TX2", etc., for batch text files and extensions ".DA1", ".DA2", etc., for the corresponding binary files.

3.6.2 Revision of Project Data

Project data, which can be expressed in the batch text file, includes all data except plan-specific data. Plan-specific data includes only fixed start-times, serialization (quasi-precedences), and fixed durations. A maximum of five sets of plan-specific data can be saved in a binary file as five plans (see section 4.1).

Often, however, scheduling points out the need for more resources or a different work breakdown (definitions of activities). Such changes are more profound and must be supressed as different versions of the project, rather than simply as different plans. To make such changes, decide on a name for the revised project; decide on a name for the binary file of the revised project; edit the binary file name and other changes into the batch text file; change the name of the batch text file; use the special utilities for deleting resources and activities, if necessary; save the new batch data file; save the new binary data file; and return to scheduling.

4.1 Plans in SARA

A plan in SARA is one of up to five different schedules for a project. Scheduling changes by the user are always performed on a specific plan - the one that is loaded.

The plans for a project are numbered 1 through 5. Plan one is traditionally called the minimum-duration plan and is identified on the screen as "1=MIN". Similarly, plan 2 is traditionally called the nominal-duration plan and is identified on the screen as "2=NOM". Plan 3 is called the maximum duration plan and identified as "3=MAX". If the user follows the traditional convention, the shortest-duration feasible plan will be saved as plan 1, the plan that is considered most satisfactory overall will be saved as plan 2, and the plan that has the longest reasonable duration (which often means the least implementation costs or resource consumptions) will be saved as plan 3. Plans 4 and 5 are not associated with any traditional meanings.

All plans for a given project have the same project data, including resource availabilities, key activity durations, consumptions of resources as a function of activity durations, and precedences. In addition, the user can optionally fix the start time and/or the duration for any activities for all plans by including fixed start times and/or fixed durations in the batch text data file for the project. Usually, however, these are free to vary from plan to plan.

Within the scope allowed by the project data, different plans represent different approaches to seeking a better schedule. For example, it may be wise to hurry up an activity so as to take advantage of a briefly plentiful availability of the resources it consumes; on the other hand, it may be wise instead to try giving the same activity a long duration so as to use only a small amount of a scarce resource at any one time.

To express various strategies and obtain various plans, the user has three scheduling actions available in SARA: start-time fixing (4.2.1), duration fixing (4.2.2), and serialization (4.2.3).

Before the user can schedule in SARA, there must be an active plan. In none is active, one must be loading by selecting "Load plan" on the schedule screen and giving the integer number of the desired plan, from 1 to 5. Once the available scheduling actions have been used to the user's satisfaction, the completed plan can be saved by selecting "Save plan" on the schedule screen and entering an integer in response to the "SAVE WHICH PLAN" prompt. The prompt includes a list of all existing plans. If the number entered is the same as one listed, the old plan of that number will be replaced by the current plan. There is no way to remove a plan, only to write over it.

4.2 Scheduling Actions in SARA

To schedule a project means to adjust start times, sequences and durations of activities, so as best to meet scheduling objectives. Usual objectives are to achieve the earliest project completion while maintaining resource feasibility.

The user does scheduling in schedule mode, which is entered from the entry screen after a binary file has been opened. If a binary file exists, the first step in getting ready to schedule is to select "Open a binary file" on the entry screen, and to enter its name, including its ".DAT" extension. Then select "Schedule mode." If a binary file does not exist, select "Batch data mode" and prepare a binary file from a text file as described in Chapter 3.

The three major functions in scheduling - start-time fixing, duration fixing, and serialization - are described in the following sections.

4.2.1 Start-time fixing

The start time of an activity is the period at which it begins. This time can be set at any time such that the activity does not extend beyond the due date (DF, or desired finish). To fix the start time of an activity, select the SF command in the left window. Select the name or bar of the activity whose start time is to be fixed. This becomes the active activity. Its name is highlighted with a box, all of its predecessors are labeled with a P to the right of their name, and all its successors are labeled with an S. Quasi-predecessors (see paragraph 4.2.4) and quasi-successors are labeled with SP or SS.

At the same screen height as the activity's bar and name, select the time period at which the activity's start is to be fixed. The Gantt screen will then show the consequences: the activity will be deactivated; a fixed start tag resembling an "L" will appear to the left of its name; its bar will be moved horizontally so as to start in the selected period; its successors and quasi-successors will move accordingly, if their start times are not fixed; negative and positive slack indicators will be revised; the time infeasibility and resource infeasibility numbers will be revised; the resource profile, if any, will be revised; and the resource conflict graphs, if any, will be revised.

To free activities so that their start times are no longer fixed, select the-SF command, and the activity whose start time is to be freed. All the scheduling consequences, as listed above, will be shown.

The SARA scheduling heuristic (see section 6.4) provides suggested start fixes for activities, in an attempt by the program to identify a resource- and time-feasible schedule. To see these suggestions displayed in shadow bars on the Gantt chart, select the HEU command. To accomplish all the suggested time-fixes in one act, accept the suggestions by selecting the ACC command.

To aid the user in making fix and unfix decisions; the Gantt chart screen shows many informative items:

Positive and negative slack indicators

Resource conflict graphs

Resource consumption/availability profile

Time and resource infeasibility numbers

A positive slack indicator is a dotted line extending to the left or right of an activity bar that shows how far the activity could be moved to the right or left (by start time fixing or other scheduling acts) without causing time infeasibility. Thus, for example, if an activity could be scheduled later, without pushing its successors beyond the due date, a positive slack indicator will be shown to its right.

A negative slack indicator is a solid line extending to the right or left of an activity bar that shows how far the activity needs to be moved to the right or left to restore time feasibility. For example, if an activity's start time is fixed N periods before the next period after its predecessor's finish time, a negative slack indicator will extend N periods to the right. Negative slack indicators are often useful in scheduling to a due date because every activity whose finish time is inconsistent with meeting the due date will have negative slack indicated to its left.

The three character codes for all resources are listed to the upper left of the Gantt chart screen. Selecting one of these resources causes the resource conflict graph to appear below the Gantt chart. A conflict graph is a bar of varying height, whose height in any period represents the relative amount by which total use of that resource by all activities exceeds availability. The height is zero if there is no overconsumption, and up to six pixels if the overconsumption is twice the resource's greatest infeasibility. The activities that are causing overconsumption of a resource can often be identified as the ones whose start or finish times coincide with steps up or down in the conflict graph. The resource code and name for each conflict graph is shown to its left.

For any one resource, a resource consumption/availability profile may be shown above the resource conflict graph by the user's selection of the appropriate resource's conflict graph (or its code or name to the left, but NOT its code in the upper left corner of the screen). Selecting a diffferent conflict graph replaces the current profile, if any, with the new one. Reselecting a resource's conflict graph whose profile is currently shown causes the profile to disappear.

The solid line in the resource profile shows the total relative consumption in each time period for the resource. The dotted line shows the availability. If the availability cycle of the resource has length 1, there is overconsumption in each period where the solid line is higher than the dotted line. If the cycle is larger than 1, there is overconsumption when the area of the solid line above the dotted line is greater than the area of the solid line under the dotted line in the cycle.

The time infeasibility number, shown as "ti" at the lower left of the screen, indicates the amount of time infeasibility in activity-period units. That is, when one activity violates a precedence, quasi-precedence or due date by one period, ti is increased by one. The resource infeasibility number, shown as "ri", indicates the relative amount of overconsumption of resources in units such that if one resource is overconsumed by 100% of its largest availability in one availability cycle, ri is increased by one.

4.2.2 Duration Fixing

To fix durations of activities, it is necessary to exit the Gantt chart screen and to select "Change scheduled activity durations" on the schedule screen. A list of the activity numbers and names is displayed. Select an activity, and the MIN, MAX, NOM, and current durations of the activity are displayed. Enter the new schedule durations. This new duration must be within the limits of MIN and MAX. A carriage return will leave the current duration as it is. Select "EXIT" to return to the schedule screen and "Schedule" to return to the Gantt chart and view and exploit the consequences of the duration change. Activities whose durations have been changed from the NOM duration have a duration fix tag displayed to the left of the name. This duration tag is a "lazy C", or a left bracket rotated ninety degrees counterclockwise. If there is also a start fix tag, the duration fix tag will be superimposed on it, giving a combination that looks like an "L" with a serif on its right foot.

4.2.3 Serialization

When resource overconsumption occurs because too many activities are using the same resources at the same time, a more flexible scheduling tool than start fixing is serialization (see section 1.2.5), which spreads resource usage over time by introducing quasi-precedences.

To specify a serialized sequence of activities, it is necessary to exit the Gantt chart and select "Serialize activities" on the schedule screen. A list of activity names and numbers is displayed. Touch an activity to activate it, if none is active. If one is active, a reselection of the active activity deactivates it, while the selection of another activity establishes the second activity as a serial quasi-predecessor of the active activity. Selection of a previously defined quasi-predecessor of the active activity deletes the relationship. When an activity is active, the cursor returns to it after selection of any other activity. When an activity is active, all of its serial predecessors are tagged to the right with "sp".

To serialize activity one before activity two before activity three, select activity three, then two, to establish two as a quasi-predecessor to three. The cursor returns to three. Reselect three to deactivate, activate 2, and select one to establish the one-two relationship, and the series is created.

Any quasi-precedence relationships that would cause a precedence cycle (i.e., any activity being a predecessor or quasi-predecessor of itself) is prevented by SARA. Since the cursor automatically returns to the active activity, and since selection toggles (reverses) the relationships, care must be taken. To review the quasi-predecessor relationships, deactivate the active activity and activate each activity in turn to display its serial predecessors, taking care to use the arrow keys only when no activity is active. (This prevents unintended alteration of serial relationships.)

4.3 Screen Control Using SARA

The schedule screen is the entry screen for scheduling acts, accessed by selecting "Schedule mode" on the SARA entry screen. The schedule screen menu is as follows:

1) Load plan
2) Save plan
3) Serialize activities
4) Set time boundaries
5) Set desired start and finish
6) Schedule
7) Set screen movement constant
8) Change scheduled activity durations
9) Change screen sequence of activities
10) Exit to entry screen

The screen also shows the name of the binary file that has been opened and the name of the plan that has been loaded. "No plan is loaded" appears upon first entry to the schedule screen, and the user must select "Load plan", and provide a plan number from among those listed. If no plan is loaded, attempted selection of scheduling acts will be ignored.

To access the Gantt chart, select "Schedule". The main window contains activity bars, with activity names to the left. The activity bar begins at the activity's start period and extends to the activity's finish period.

The appearance of the Gantt chart screen can be controlled by setting time boundaries and desired start and finish dates of the project. The desired project start date controls the earliest start of activities that have no predecessors or quasi-predecessors, and the desired project finish date controls the due date against which positive and negative slack are determined. The user may wish to delay the desired start date as an alternative to fixing all start times, or may desire to make the due date temporarily earlier to identify the most critical activities with positive slack indicators.

To set desired start and finish times, select "Set desired start and finish" on the schedule screen. If the NORMAL calendar option is in effect, enter the new times as integers or NORMAL codes (see section 1.4.2). If the WORKDAY option is in effect, enter the new times as WORKDAY codes (see section 1.4.2).

NOTE: Desired start and finish times are network data that apply to all plans.

Time boundaries determine what span of periods will be shown on the Gantt chart at once. To set time boundaries, select "Set time boundaries" on the schedule screen menu and enter the display start and the display end in the same manner as start and finish times were entered.

The resource conflict graphs and the resource consumption/ availability profile use up space at the bottom of the Gantt chart, so the user may want to eliminate unnecessary conflict graphs or the profile, in order to see more activities at one time. The user may also want to adjust the screen movement constant to control how many activities are rolled on or off the screen by the UP and DWN commands. To bring N activities into view that are presently below the screen, select the UP command. To bring into view N activities above the Gantt chart, select the DWN command. N is the screen movement constant, which can be changed from its default value of four by selecting "EXIT" command, to go to the schedule screen, and then "Set screen movement constant". Then enter a new integer number of lines and return to the Gantt chart by selecting "Schedule".

The user may also desire to change the screen sequence of activities so that the set of activities currently displayed are those currently of interest. To do this, select "Change screen sequence of activities" on the schedule screen. The list of activity names and numbers will be displayed. Activate an activity and select another activity. The active activity will now be after the selected activity. The list of activities is rewritten showing the new order; this order will be the one on the Gantt chart. The cursor is left at the new location of the previously active activity, and no activity is active. This can be repeated as desired. This input protocol obeys the metaphor "put this one after this one."

5 SARA Reports

Reports can be prepared for a project described by an open binary file by user selection of "Report mode" on the entry screen. Each set of reports covers one specific plan (schedule). Therefore, upon entering the report screen, select "Load plan" and enter the number of the desired plan.

Four reports are available: a schedule table report, resource table report, Gantt chart report, and resource chart report.

When a report is created, SARA computes the report data and appends it to a file of name "filename.RPT", where "filename.DAT" is the name of the binary file from which the report was prepared.

5.1 The Schedule Table Report

To create a schedule table report, select "Schedule table report" on the report menu.

The first line of the report will be the general title "*ACTIVITY SCHEDULE REPORT", followed by the filename of the open binary file. (The specific plan number is not given.) The two line description of the project as defined in the batch file comes next. The body of the report lists each activity and the following information: its scheduled start, finish, and duration; its scheduled consumption of each consumed resource, including its total consumption and the rate of consumption at the scheduled duration; and the "key data" - MIN, NOM and MAX durations - and the total consumption of each consumed resource at each of these durations. If an activity is critical (has zero or negative slack), it will have a "=crit=" tag to the right of its name. The rest of the report contains the "general precedences" - a list of each activity's predecessors and quasi-predecessors, if any.

5.2 The Resource Table Report

To generate a resource table report, select "Resource table report" on the report menu.

The first line of the report contains the title "RESOURCE SCHEDULE REPORT" and the name of the binary file from which it was prepared. The specific plan number is not listed. The two line description of the project follows. The body of the report lists each resource's availability and consumption data, and the consumption rates by each activity of that resource.

5.3 The Gantt Chart Report

To prepare a Gantt chart report, select "Gantt chart report" on the report menu.

The Gantt chart consists of a time scale bar chart. If an activity is critical, its bar is represented as a string of equal signs ("="). Non-critical activities' bars are shown as a string of esses ("S"). Serial period numbers are given at the bottom of the chart, and period codes are given at the top.

5.4 The Resource Chart Report

To generate a resource chart, select "Resource chart report" on the report menu.

The resource chart is a timescale display showing each resource and its availability, consumption, surplus and defenciency in each time period.

5.5 Report Generation and Control

Any number of reports can be generated from SARA without leaving report mode. After each selection, there is a wait while SARA performs the necessary calculations and stores the results. It is a good idea to put reports for the same plan together, though a series of reports from different plans can be created by loading a new plan in between report selections. Paper records should be kept of which plan pertains to, as the plan number does not appear on the report.

Every report created for every plan of a project is appended to the report file in the order in which they were created.

To view a report, the user must exit SARA and either "type" the file under DOS or view it through EDLIN. In printing a report, an 80-column printer is assumed. If a report exceeds 80 columns, such as the chart reports, which are of arbitrary width, SARA "folds them over" in such a way that the resulting output can be cut and pasted into reports of the proper width.

6.1 Hyperbolic Interpolation

Consider the following scenario:

"A job requires 12 man-days of computer programming, plus 2 man-days of familiarization work by each non-supervising programmer, plus a full-time programmer to do supervising for the job's duration."

This scenario implies a specific model of resource consumption as a function of duration. If we let v represent the effort in man-days by non-supervising programmers, and let w represent the effort in man-days by the supervisor, then the total effort is $y = v + w$. If the duration is x days, a direct translation of the scenario statement is

$$y = 12 + 2v/x + w \tag{1}$$

which which $w = x$ and $v = y-w$ leads to

$$y = 12 + 2y/x - 2 + x \tag{2}$$

or

$$y = (10 + x)/(1 - 2/x) \tag{3}$$

which defines a hyperbola with a vertical asymptote $x = z$, an oblique asymptote $y = 10 + x$, and a center of $(2,12)$. Figure 6.3-1 shows the relationship of total effort y, in man-days of programming and supervision, to job duration, for durations from 3 to 10 days.

We see both from the scenario statement ("...plus 2 man-days of familiarization work by each...") and from Equation 3 ($1 - 2/x$ denominator) that the meaningful domain is $x > 2$. For $x \leq 2$, nothing but familiarization work would be done and Equation 3 would yield the lower branch of the hyperbola.

The duration x that minimizes effort can be found in the region $x > y$ by setting $dy/dx = 0$ in Equation 3:

$$x = 6.8990 \text{ days}$$

giving the minimal effort $y = 23.7980$ man-days.

The <u>consumption</u> <u>rate</u> of manpower can be defined as $z= y/x$, which has units of man-days per day and simply represents the number of programmers and supervisors. In terms of the rate z, Equation 3 becomes

$$z = (10 + x)/(x - 2) \qquad (4)$$

This formula defines a right hyperbola with a vertical asymptote $x = 2$, a horizontal asymptote $z = 1$, and a center-to-focus distance of 3.4641. Figure 6.3-2 shows the relationship of consumption rate (number of people) to job duration.

The simplicity of Equation 3 or 4 as a model of the comsumption/duration relationship given in the above scenario brings up the possibility of a more general form of the model. Parameterization of Equations 2, 3 and 4 gives the following equations:

$$y = c_1 + c_2y/x + c_3x \qquad (5)$$

$$y = (c_1 + c_3x)/(1 - c_2/x) \qquad (6)$$

$$y = (c_1 + c_3x)/(x - c_2) \qquad (7)$$

These equations give three descriptions of a hyperbolic consumption/duration model that has three parameters: c_1, c_2 and c_3. Equation 5 can be called the "modeling" form. It shows that the model expresses resource consumption as a sum of three terms: an intrinsic work content c_1, an additional component of work that is proportional to the rate of work, and a third component of work that is proportional to the duration. Equation 6 can be called the "consumption" form of the model, and Equation 7 can be called the "rate" form.

The consumption form, Equation 6, defines a hyperbola with vertical asymptote $x = c_2$ and oblique asymptote $y = c_1 + c_3x$. The rate form, Equation 7, defines a right hyperbola with vertical asymptote $x = c_2$ and horizontal asymptote $z = c_3$.

6.2 The SARA Scheduling Heuristic

SARA has a scheduling heuristic that attempts to find a resource- and time-feasible schedule of minimum duration. This heuristic can be called for on the Gantt chart by selecting the HEU command, whereupon SARA computes the schedule and reports it in the form of shadow bars.

If the heuristic recommendations are accepted by the user through the selection of the ACC command, start times of activities are fixed for those activities that the heuristic recommends should be started after their respective early start times.

The heuristic does not try serialization or duration changes.

Here is how the heuristic works:

For each activity that has not yet been given a trial start time and has no predecessors that have not been given a recommended start time, the heuristic in turn tries various start times, starting with the earliest possible, until the activity can be scheduled. Then the activity becomes one that has been given a trial start time. Its consumptions are subtracted from availabilities, and another activity is tried. When all the activities have been given a start time, the heuristic stops.

If an activity cannot be scheduled at any time without causing resource infeasibility, it is scheduled at its early start time, and the heuristic goes on to the next activity.

After acceptance of a new schedule, the resource infeasibility number "ri" will be zero, unless one or more activities were impossible to schedule without infeasibility. If this happens, the user can examine the schedule, change durations, and ask for the heuristic again. In most cases, it is also desirable to free the start times of activities that were fixed by earlier acceptance of a heuristic before asking for another, especially those activities that are closely related to an un-schedulable activity, either by successorship or quasi-successorship or competition for the same resources.

6.3 Error Messages in SARA

All screens except Schedule Mode:

"Chosen file does not exist, reenter file name." - Often caused by neglecting the ".DAT" or ".TXT" extension to an otherwise valid file name. To view all valid file names it is necessary to exit from SARA into DOS.

"Chosen file already exists, choose another file name." - SARA prevents writing over any file except the default file name in the batch data file for a project.

"Newly chosen name exists, choose another file name." - (same as above)

"No active file to modify, select a file before editing." - A file must be loaded before any other work can be done.

"There are no activities to schedule in schedule mode." - SARA cannot operate on a file with no activities.

"The file description cannot be changed without an active file." - A file must be loaded before any other work can be done

Batch Data Mode:

"The batch data file has an error on or before line N." - This message identifies the first line at which a discrepancy wa detected. Common causes: The general area of the batch text fi has the wrong number of lines; the project description must occupy two lines, and there are two extra lines if the calendar choice is workday. The resource area has the wrong number of lines; after each resource code and name, there are exactly thre lines before the availability segments. The END OF RESOURCES line, END OF ACTIVITIES line, END OF CODE WEIGHTS line, or END O FIXES line is missing or malformed. There was an attempt to specify more than one indirect resource consumption or more than one code weight pair on a single line.

Report Mode:

The prompt shows the available path [illegible], and enter it as an integer from 1 to 5.

Schedule Mode:

"REDUNDANT SELECTION OF AN ACTIVITY" - Review the legal touch sequence for the input act being attempted.

"ACTIVITY MUST HAVE A FIXED START TO BE FREED FROM A FIX" - The free-fix command is active (it appears as "SF" overstruck with hyphens); free only the activities marked to the left of their names with the "L"-shaped fix-tag, or activate a different command.

"SCREEN ALREADY INCLUDES ALL THE ACTIVITIES" - UP or DWN has been selected when there are no activities out of view.

"INVALID LOCATION FOR ANY ACTION TO BE TAKEN" - A selection has been attempted which cannot be interpreted by SARA. Review selection procedures for the input act being attempted, or place the cursor more accurately.

Menu Items on Main Entry Screen:

Entry Screen

1) Batch data mode
2) Schedule mode
3) Report mode
4) Open a binary file
5) Save current binary file
6) Change file description
7) End session

Menu Items on Batch Screen:

<u>Batch Screen</u>

1) Load batch data
2) Save batch data
3) Save binary data
4) Delete activities
5) Delete resources
6) Return to entry

Menu Items on Schedule Screen:

Schedule Screen

1) Load plan
2) Save plan
3) Serialize activities
4) Set time boundaries
5) Set desired start and finish
6) Schedule
7) Set screen movement constant
8) Exit to entry screen

Menu Items on Report Screen:

Report Screen

1) Load plan
2) Schedule table report
3) Resource table report
4) Gantt chart
5) Resource chart
6) Exit to entry screen