PROKON/PLAN-A MODELLING TOOL FOR PROJECT PLANNING AND CONTROL [1]

TRYGVE M. H. REENSKAUG
Central Institute for Industrial Research
Oslo 3, Norway

The yards of the Aker Group in Norway have been using network techniques and ad hoc methods for planning and control for many years. The systems used have not been satisfactory, and a new moelling tool was devised for the yards. The basic building block of this tool is considerably more powerful than the activity of the traditional networks. It is called a Component, and carries both its own data and the algorithms needed to manipulate them. Each Component represents an identifiable portion of the real world in the yard. The planning process consists of the Components sending information to each other (messages), and acting upon information received. An experimental system for the planning and control of the Design Sector of A/S Bergens Mekaniske Verksteder has been developed.

1. INTRODUCTION

The Central Institute for Industrial Research and the Aker Group of shipyards have pooled their resources for advanced R&D since 1960. In the first decade of this cooperation the emphasis was on the development of systems for the computer-aided design of ships, one result being the Autokon system which is now being used by shipyards all over the world.

Although there still remains a great deal to be done in the technical sector of shipyard operations, we have for some years felt that there are also pressing problems in the planning and control sector of large manufacturing projects. This need was evident in the good old days when shipbuilding was a proficable business, but is even more so today when the Aker Group is switching part of its capacity to off-shore construction.

The yards of the Aker Group have for many years been using network techniques. In addition, they use some specialized systems for parcicular parts of their production. The yards felt that these systems were not quite sacisfactory for their needs, and a study was undertaken to analyse the situation and, if possible, suggest a suitable solution.

2. THE STUDY AND THE PROPOSED SOLUTION

Our study disclosed three basic sources for the yards' dissatisfacticn with existing systems:

- There were some clear weaknesses associated with the parcicular network program used by the yards.
- The basic principles behind all netvork techniques strictly limit the types of models that can be implemented.
- A large portion of the planner's vorking day is spent on unstructured activicies like general coordination and trouble-shooting, activities that receive no support from the traditional computer tools for project planning and control.

2.1 Weaknesses in the particular network proram

The yards had listed a number of modifications and expansions that they wanted made to their network program. They needed better facilities for working on several projects simultaneously, better tools for handling sub-networks, some new output listings, better commands for updating and correcting an existing network, better facilities for handling progress reports and so on.

2.2 General limitations in the basic network techniques

All network techniques are for the planning of <u>projects</u>. The yards use activity networks, where any operation that must be done, is called an activity. An activity is characterized by its technological dependencies on its <u>predecessors</u> and <u>successors</u>, and by its <u>duration</u>. This simplicity in its fundamental building block is the secret behind the success of the network technique.

(i) Network programs must be "tricked"

Where network techniques are employed, we find numerous instances where the planners have to exceed the limitations of these techniques. With varying success, they try to "trick" their programs in order to get the desired results.

One example is that they have to choose between an activity-oriented network and an event-oriented network. In the yards, most needs are best met through an activity oriented network, but there are certain critical events (milestones) that must be closely controlled.

1. PROKON is a registered trademark of the Central Institute for Industrial Research.

(ii) Planning of resources may be particularly difficult

The planning of resources is not part of the basic network technique, but most network programs have added some facilities for the automatic summation of resource requirements or even adjustment of activities according to available resources.

This is not satisfactory for the planning of all shipyard resources. In the plate cutting and assembly shops, for example, the main problem is the proper utilization of widely different resources such as cutting machines, floor area, and semi-automatic block assembly lines. In the block assembly line, a large block may not be followed by another large block. The proper planning of floor-space utilization is a two-dimentional problem requiring special algorithms. In a pipe manufacturing plant, attempts should be made to group similar operations into batches in order to minimize machine tool set-up time. The technological dependencies between activities are still present, but appear only as wide limits within which the planning has to be done.

2.3 Conununication with the outside world

It may seem paradoxical, but a planner uses a relatively small portion of his time grappling with activity times and resource loading. It is planning in a much wider sense of the word that fills his day (and sometimes his nights too).

The planner has a large and important net of contacts to all parts of the company, its suppliers, sub-contractors and customers. Information must be collected and discerned, misunderstandings must be cleared up, progress of important activities must be chased, reports must be written, questions must be answered etc.

Some of the planner's contacts are of an informal kind, but the main bulk of the information is transferred through the formal information system. Often, data processing systems are found at both the sending and the receiving end, but the communication between them is still manual.

2.4 Solution: A more powerful principle needed for planning and control

The basic network model seems too simple for the satisfactory modelling of large construction pr jects. Further, the proper loading of resources is a far more complex problem than recognized in commercially available systems.

Production bottlenecks and the proper methods for planning them, vary with time. Traditional planning systems operate according to a predefined set of rules. We feel that any attempt at developing such a rigid system for the Aker Group would be doomed beforehand. A successful system must have natural openings for the simple inclusion of new algorithms for solving new problems as they arise.

Our analysis led us to the conclusion that the yards' needs could not be satisfied by merely making a new and better network program; we needed a new modelling tool that should supersede the old network model. This tool should overcome all the above deficiencies in a "natural" manner, while retaining the basic simplicity of networks.

(i) Our solution: a game model

Our choice fell on a model where the planning is performed as a game between two or more players, each player being responsible for some part of the total problem and being very knowledgeable about that part. A simple illustration is shown in fig. 1.

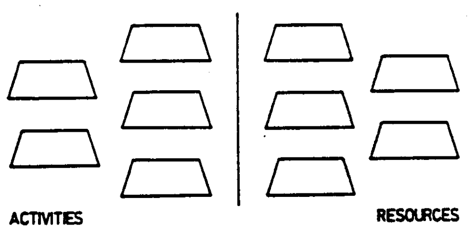

Fig. 1. Planning and control as a game between two sets of players: Activities and Recources. The responsibility of each Activity player in the game is to ensure that each activity is completed on time, while the responsibility of each Resource player is to ensure the "best" possible utilizacion of that resource.

Somewhat more formally, we will say that we decompose the complete planning and control system into a number of Components. Each Component is assigned the responsibility for a clearly bounded part or the whole planning and control process. It is furnished with the necessary algorithms for performing its function, and keeps all necessary data about its part of the real world in a local file. Logically, each Component may be represented by its own little computer with all relevant data and procedures.

The planning and control process takes place through the Components exchanging information by sending Messages to each other. A Component receives a message from another Component, interprets this message and performs the necessary actions. The overall structure of the process is determined by the general pattern of message flow and the contents of the messages, while the details of the process are determined by the algorithms in the various Components.

(ii) Discussion of the solution

EQUALITY. Our model has no built-in priority betweeen the various factors that influence the planning process. Any significant factor will be represented by a Component, examples being the Activity and Resource Components mentioned in fig. 1. If there is a tight time schedule, the activities will dominate the planning game. On the other hand, if the resources are particularly short, they will dominate. The balance between activity and resource domination may vary between different parts of the same plan.

PERSONALITY. Since the algorithms for treating a given message are associated with the receiving Component, the Component appears to have "personality". The Component representing the resource "welding dept" may perform resource allocation in one way, while the Component representing the "pipe fitting dept" may do it in some other way.

EXPANDABILITY. Let us assume that our planning system has been in operation for some time. We then find that we need to include the resource "big crane" into our planning model. Let us further assume that for some reason, this resource can not be planned according to any of the rules for resource planning that we already have in our system. We must then create a new kind of Component that we call "big crane". We provide it with suitable programs for treating all messages that this Component may receive, and add the new Component to our running system. No old programs are changed, the complexity has not changed significantly, and all the people who are not particularly interested in the big crane need not even know it is handled in a special way.

DISTRIBUTION. Our model is based upon Components that send and receive Messages. Within our general framework, it is clearly possible to let the Components of our planning system send messages to, and receive messages from other systems. Examples of such systems are the design system, the purchasing system, the materials management system, the accounting system, etc.

We assert that our model has sufficient power to map all the formalized planning problems we can see in the engineering and construction of large and complex steel structures.

3. DEVELOPMENT OF A PROKON SYSTEM FOR PLANNING AND CONTROL

The development of a PROKON planning system consists of four steps:

- The definition of the commands to be made available to the user for his control of the system.
- The decomposition of the total planning problem into a number of distinct Components.
- Determination of the Component interaction needed to obtain the desired effects of each command. (The rules of the game.) This entails both the definition of the types of Messages needed in the system, and the pattern of Message interchange.
- Programming of the algorithms that determines the "personality" of a Component. For each Component type, an algorithm must be written for each type of Message that it may receive.

Development and maintenance of a planning system usually necessitate going through these steps many times in an iterative fashion.

We have developed a programming aid that is of use during both development and maintenance of PROKON planning systems. This aid, which we call PROKON/PROG, utilizes the propietary data base system SIBAS. New features of the planning system may be fed into the computer as they become known, and changes to previously defined features may readily be introduced.

4. AN EXAMPLE OF A PROKON SYSTEM FOR PLANNING AND CONTROL

A/S Bergens Mekaniske Verksteder (BMV), a company within the Aker Group, was modifying its design sector into a matrix organization. Thorough discussions went into defining the responsibility and authority of the new organizational units. Some highlights of the definitions which are pertinent to the planning and control system are shown in fig. 2.

4.1 Overall system

A PROKON system for the planning and control of the design sector was developed. The design was done on two levels of detailing: first, an overall design of the complete system, and then a detailed design of each of its parts.

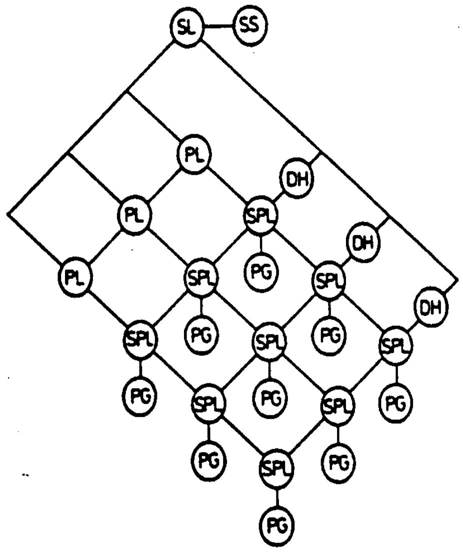

Fig. 2. The matrix organization of the Design Sector of A/S Bergens Mekaniske Verksteder.

SL: SECTOR LEADER is responsible for all operations within the Design Sector.

SS: SECTOR STAFF assists all parts of the Design Sector with planning, reporting computer utilization, etc.

DH: DEPARTMENTAL HEAD is responsible for maintaining a high overall standard of technical know-how within his department, and of an efficient utilization of the total resources of his department. (Piping design, electrical design, etc.)

PL: PROJECT LEADER. Responsible for the establishment of a design project, the definition of its major activities with technological dependencies, time constraints, resource requirements, accounting procedures, etc.

SPL: SUB-PROJECT LEADER is given responsibility for completing a set of major activities to a high technical standard within the given time constraints.

PG: PROJECT CROUP consists of the designers of a department that work on a particular project.

In the overall design, a PROKON Component was defined for each of the organizational units of fig. 2 except for the sector staff. These Components are shown in fig. 3 together with the major types of information flowing between them. Note the one-to-one correspondence between the human organization of fig. 2 and the formal information system of fig. 3. In our opinion, such a correspondence is a prerequisite for obtaining a proper user understanding of an information system.

The PROKON model of interacting Components, each Component with clear responsibility for part of the total planning problem, seemed to be readily appreciated by users, and the definition of the Components and the discussion of the exact nature of the information flowing between them, proved to be a fruitful viewpoint in the process of developing the new organizational plan.

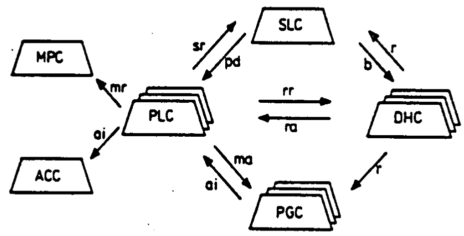

Fig. 3. Overall system design consisting of a Component for the Sector Leader (SLC), one Component for each project (PLC), one for each Head of Department (DHC), and one for each project group (PCC). Further, two Components that belong outside the Design Sector are shown: A Component representing materials procurement (MPC) and one representing accounting (ACC). The major data flows are shown in the figure:

pd: project definition
sr: summary reports
r: resources
b: budgets
rr: resource requirements
ra: resource allocations
ma: major activities
ai: accounting information
mr: materials requirements

The interrelation with organizational units outside the design sector is also illustrated in fig. 3 with the communication paths to the accounting sector and the materials procurement sector. In fact, fig. 3 may be thought of as a detail ot a much larger picture, that of the company's overall formal information system.

4.2 Details of the subsystems

After the organizational plan was finalized, our task was to develop a working information system according to the overall model of fig. 3. It was decided to use computerized systems for PLC, the project leader's Component, and DHC, the departmental head's Component. The sector leader's Component and the project group's Coponent should be realized as manual systems for the time being.

We have chosen the project leader's Component to illustrate the decomposition of a major Component. A good starting point for the decomposition of this Component is the project leader's responsibilities. A detailed diagram of this Component is given in fig. 4.

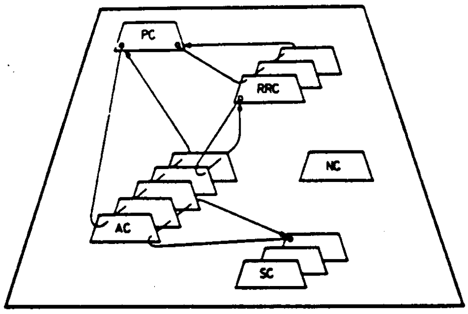

Fig. 4. Details of the project leader's Component (PLC).

PC: PROJECT COMPONENT. Represents the complete project in the computer system.

AC: ACTIVITY COMPONENT. Represents one of the project's activities. Belongs to the PC.

RRC: RESOURCE REQUIREMENT COMPONENT. Represents the project's total need for a given resource. Belongs to the PC and owns the ACs needing the resource.

SC: SUBJECT COMPONENT. Economic reporting serves many purposes: Finance, billing, budget control, etc. A Subject Component serves as a clearing centre for such reporting, distributing the raw data from time sheets, etc. to the various accounts.

NC: NETWORK COMPONENT. Represents a standard program for netWork planning.

The next steps in the design are to determine the user commands needed by the project leader (or his staff) in order to define and manipulate his plans, to define all the types of messages and patterns of message flow needed to realize each command, and finally to program the required procedures.

In fig. 5, (next page) we give an example of the pattern of messages used for resource allocation. Apart from this, we will not show any details of this planning system since it has been tailor made for a particular situation. However, it may be of interest to record that the design of the system started in February 1976, the programming started in March, and the system was completed and ready for installation in April 1976.

(i) Utilization of standard software packages

In fig. 4 there is an additional Component called a Network Component, NC. At certain points in the process of making his plans, the project leader may want to employ standard network techniques. Rather than developing yet another computer program for this purpose, we decided to employ an available standard packet. Whenever the project leader gives the command:

- NETWORK CALCULATION

NC arranges for network planning to be performed. It generates a simulated input card deck for the standard network program and asks the computer operating system to start it. When the network program has finished its job, NC picks up the internal representation of that program's output listing before it hits the printer. NC interprets this listing, distributing the results of the network calculation to the various AC's.

It is important to stress that we have not introduced the slightest modification to the standard network program. There will therefore be no special difficulties with the maintenance of that program nor with the responsibility for its correctness. It cost us less than a week's work to give the project leader this well thought out and highly reliable facility.

We believe that this little experiment points towards a new and very efficient way of tailoring the services of standard, general-purpose packages into special-purpose tools for individual users, and we expect to see the method widely adopted in a great number of different circumstances.

5. CONCLUSION

It seems that our way of modelling systems for project planning and control is applicable to a large class of problems hitherto only amenable to solution by special ad hoc methods. Our methods contain the network methods as a subset, and we can easily develop a PROKON system performing network planning. We do not recommend this because excellent network programs are available, and a PROKON system for the same purpose will need considerably more computing facilities due to its greater generality.

PROKON shows its power, however, in complex situations where standard methods have to be stretched uncomfortably far, or where one has to use several different programs for planning different parts ot an integrated project. Further, PROKON supports the cooperation between groups of humans in an environment of distributed competence and authority, and it facilitates the development of a specialized, unified tool for each of the user groups regardless of the variety of the standard programs that are needed to back up the operation of that user's Component.

ACKNOWLEDGEMENTS

The work presented here has been financed through grants from the Aker Group and the Central Institute for Industrial Research. All through the project there has been a very fruitful interchange of ideas between the staff of the Aker Group and the Institute. Particular thanks to Messrs. Sven Ytrehus, O.J. Heldal and Sven Milch of A/S Bergens Mekaniske Verksteder and A.H. Landmark and U. Brox of the Central Institute for their valuable contributions to the project.

The basic ideas behind the PROKON model stem from the Simula Language [1] and the RC-4000 monitor [2]. The reported project is part of a wider research program searching for better structures for a company's over- all information system [3]. Similar models are studied in many other places, particularly as applied to operating systems, etc. (See for example [4].)

The command: RESOURCE LOADING is given by the user and is translated to message of type 140.

Tbe Project Component (PC) selects the first resource to be loaded. (Resources are here loaded sequentially. Necessary backtracking is supposed to be done through manual intervention.) When all resources have been loaded, message 148 to the user indicates that the command is completed.

The Resource Requirements Component (RRC) asks the Component of the departmental head (DHC) to give the current capacity limits through message 150.

DHC returns the capacity it thinks fit to release for our project through message 151.

RRC then asks all ACs for their resource requirements and their earliest and latest start and finish.

Tbe ACs return the requested information.

RRC performs loading according to its own, built-in rules and reports the planned times for start and finish to the ACs. It also reports back to PC that it has completed its loading.

The ACs note their new planned start- and finish times.

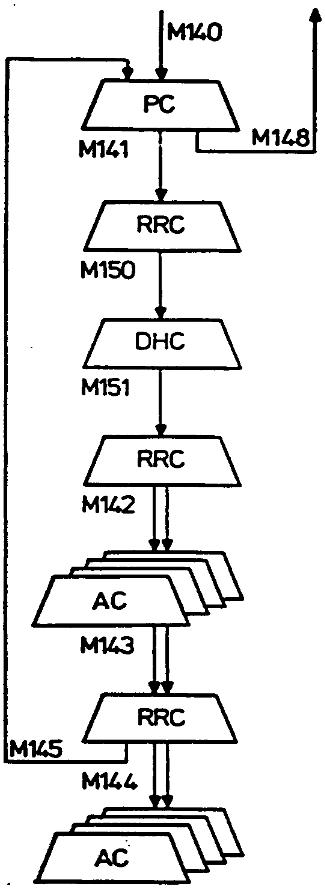

Fig. 5. This example of a resource loading algorithm illustrates the informal, movie-type of documentation used to describe the organized flow of messages resulting from a given user command.

REFERENCES

[1) Ole-Joban Dahl and Kristen Nygaard, SIMULA 67 Common base definition, Norwegian Computing Centre, 1967.

[2) Per Brinch Hansen, RC-4000 multiprogramming system, Copenhagen, April 1969.

[3) Trygve Reenskaug, Administrative control in the shipyard, ICCAS Conference, Tokyo 1973.

[4] C.A.R. Hoare, Monitors: An operating system structuring concept, CACM 17, 10, Oct. 1974.

NOTE:
This paper was originally presented at the IFIP Congress in Toronto, Canada, 1977 and first published in the 1977 IFIP Proceedings.
Reset by the author July 2003.