Programming the American Aerospace Industry, 1954-1964: The Business Structures of Technical Transactions

Glenn E. Bugos
German Marshall Fund of the United States

Program management swept through American industry in the 1950s and 1960s. The movement was championed by a new breed of program managers dedicated to the science of creating new technological systems. Their work was multidisciplinary but uni-motivational; they borrowed and experimented with any technique that served a technological system and its time economy. That in itself was not new. Thomas Hughes tells us that all systems historically depended on the fox-like and time-obsessed behavior of systems builders. What was new, in the mid-1950s, was systems builders who had professional identities as "program managers," and who urged their firms to make the planning and creation of new programs a regular part of their institutional life.

The American aerospace industry in the 1950s offered a receptive context for the development of program management. Aerospace attracted headlines and money, and thus some of the best technical and managerial minds in the world. These new program managers devised a palette of management tools that they claimed they could apply to create new aircraft surely and swiftly. They also claimed that program management best served their customers, who knew exactly what they wanted in a system but needed contractual help to make it fly. Military officers and civil servants in the Defense Department specifically encouraged firms to appoint program managers who could marshal resources and assume responsibility for their weapons systems. The Armed Services Procurement Act of 1948 broke down legal fences and allowed program managers virtually free rein. In 1961 the Air Force Systems Command adopted its 375 Series of regulations that made program management a legal requirement of defense contracts. By the 1960s, more firms needed program managers, especially those who exuded confidence.

The result was a boon for management consultancies. Unlike most of the people that comprised a modern firm--accountants, engineers, machinists--no union or professional association or university department defined the identity of a program manager. The disciplinary definition of program management thus fell to consultancies such as MacAulay Associates,

---

BUSINESS AND ECONOMIC HISTORY, Volume Twenty-two, no. 1, Fall 1993.
ISSN 0849-6825.

Arthur D. Little, McKinsey & Company, and moonlighting professors from industrial relations institutes at MIT and Caltech. These consultants studied project histories, both of weapons and civilian products, and deduced general rules, systematic sets of jargon, and bullet charts and diagrams. They then taught high-priced seminars intended to change the way technical managers thought about the relations between parts and wholes--both equipment to systems, individuals to groups, and contractors to customers--and about the importance of time in modern technology. We can, in fact, define a disciplinary core of program management by how it represented the organization of engineering. Management consultancies sold an ideal of program management, touting it as a miracle cure for all industrial ills.

Though these consultancies sold the idea of program management to a vast array of firms, it is not yet possible to do a quantitative survey of how program management changed the face of American industry (along the lines of Alfred Chandler's work). Instead, I will simply present two of the technologies of representation used in the 1950s in order to define the paradigmatic core of program management: namely the matrix organization chart and the time network chart.

The Matrix Organizational Structure

In the early 1950s the McDonnell Aircraft Company of St. Louis, like most aerospace firms, organized itself functionally. In other words, vice-presidents for sales, finance, engineering, manufacturing, and customer service all reported directly to the president. When the Defense Department asked McDonnell to build an aircraft, each functional department sent a representative to an aircraft committee which then divided and assigned the work among themselves. The sub-departments within the engineering department argued extensively, for example, about whether the structural or aerodynamics engineers needed to finish their work first, whether the hydraulics group could use a new actuator that the controls people had never seen, whether space and electric power were reserved so the electronics group could later propose a reconnaissance version. The departments then handed off a project as it reached the next step in its life-cycle. Contract types defined these steps. The Defense Department usually awarded cost-plus contracts for engineering a prototype, a separate fixed-price manufacturing contract, a cost-plus modifications contract, and finally a directly-billable service contract. Most of McDonnell's work at the time was for the Navy, which also organized its Bureau of Aeronautics by function.

A combination of external factors moved McDonnell toward a product organization. First, the Armed Services Procurement Act allowed the Defense Department to award McDonnell sole-source contracts that gave McDonnell increased control over its subcontractors (making it a true prime contractor), and more control over the aircraft throughout its lifetime. Second, McDonnell received more missile contracts, a technology new to both them and the services. Consequently, the services could not impose their previous functionally-oriented oversight on the missile contracts. Third, to compress development time, contracts often specified that production tooling and squadron training would begin before McDonnell completed the aircraft design. Known as "concurrency," this practice precluded a clean hand-off from one department to the next. Most importantly, McDonnell's customers moved toward a product-oriented structure. In 1952 the Air Force reorganized its design engineers into the ARDC (Air Research and Development Command). The Air Force then formed Weapon System Project Offices to ease the transfer of oversight from the ARDC to the Air Material Command, responsible for maintenance. Because McDonnell now answered to only one office, the Air Force wanted only one McDonnell employee answering to them. Since J.S. McDonnell, the president, spent most of his time answering to the shareholders, he designated a deputy to gather information from the departments in order to brief the Air Force Weapon System Project Officer.

This centralizing of information marked the first stage in the evolution of program management. In the early 1950s, McDonnell appointed roughly twenty "company-wide program managers," mostly engineers with expertise in testing or marketing. The program manager led the conceptual design of the aircraft and defined the contract with the customer. He then delegated portions of the contract to the departments and pressured them to do the work on time, to specifications, without billing the project contract for more materials and hours than were budgeted. The program manager monitored the program from inside the company, just as the military did from outside. The program manager often allied himself with his customer and their weapon system, rather than with the shareholders and their capital.

Although the program manager had to convince his customers that the departments were interacting smoothly, he had no epistemological authority to mediate the inevitable technical disputes. These conflicts became more intense during the 1950s as aircraft were made more complex, that is, as airframe firms tried to squeeze more electronic equipment into them. The program manager "owned" a handful of project engineers, who reported to a chief project engineer with a wide range of engineering experience and did what was later called "systems engineering." Systems engineering involved weighing the trade-offs between the design of the prototype, dividing work packages so the departments could do their work concurrently, keeping a weight and space budget for the airframe, and auditing all test results so as to keep the departments honest. When conflicts arose, only the departments had an overhead budget and sufficient manpower to engage in battle. Consequently, the program team would have to bide their time and accept a solution suggested by the departments.

In the interest of overcoming such debilitating departmental gridlock, McDonnell explored ways of modifying their corporate structure. Furthermore, in the late 1950s, McDonnell launched several spacecraft programs for NASA while the F-4 Phantom program occupied most of the company's aircraft engineers. To strengthen these program managers, in 1958 J.S. McDonnell permitted them to directly hire more engineers. This resulted in the program teams doing more "line" work and departments more "staff" work. Program managers, with the aid of more preliminary design money from their customers, drafted engineering plans which were much more detailed. The program manager personally and explicitly defined the contract. Once he set the project plans in motion, they served more as an organizational control mechanism than an inter-firm contract across departmental boundaries.

To oversee these plans and to serve the aircraft over its lifetime, the program manager had his own payroll to hire project engineers better trained in certain technologies. These project engineers served as ambassadors to the departments, to McDonnell's research and testing laboratories, and to McDonnell's suppliers. When the project engineers anticipated an integration problem, or when they simply needed more detailed work done on one part, they rented time from the pool of departmental engineers.

To punctuate the new power of the programs, in 1958 McDonnell also created the position of general manager with status comparable to that of department heads. The general manager spoke to the president of the firm on behalf of all program managers. At the same time, McDonnell launched a five-year building program to update their St. Louis campus and rationalize definitions of department capabilities. By the late 1950s McDonnell was structured as a balanced matrix organization, with neither departments nor programs defining the line product of the firm. If line and staff structure of the 1880s, with a strong product orientation, was the thesis, and a multidivisional structure of the 1920s, with organizational capabilities grouped according to function, was the antithesis, then a matrix structure was the synthesis. The flow of work and responsibility within the firm was well-described as a matrix because the hierarchies of each department crossed over the hierarchies of many program teams.

Within a matrix organization, the departments served as repositories for expertise--in manufacturing methods, purchasing, financial accounting, sales and support, testing and quality control, and the various engineering disciplines. The departments explicitly connected the firm with the environmental state of the art. Departments hired young people from university departments, created temporary research laboratories to capture patents or prestige from professional societies, sat through lengthy briefings from suppliers trying to sell them new types of components or manufacturing equipment, and drafted Comments when the Defense Department updated their material or process standards. The departments brought organizational capabilities into the firm and stood ready to sell their expertise to any program group needing help on a new type of aircraft. It became commonplace in the 1960s for the Defense Department to give general research contracts directly to McDonnell departments. These basic research contracts allowed McDonnell to accumulate expertise in a new technology while, at the same time, to make it part of the extant state of the art by publishing new material standards or research reports. McDonnell, for instance, received an Air Force contract requesting a report on the potential of using composite materials on future aircraft and missiles.

More often though, the programs served as the profit centers within the corporation, and the departments served as the cost centers. That is, the programs controlled the contract funds, and determined whether work would be assigned to the departments, awarded to an outside firm under subcontract, or whether advice would be requested from Defense Department engineers. The program managers knew which technologies they needed to apply to an aircraft but often needed help in locating expertise in these technologies. To make their expertise more accessible, the departments gave their employees increasingly detailed work descriptions, declaring each employee an expert in a particular technology. As McDonnell grew larger the departments differentiated labor into specialties; the program teams then reintegrated these specialists to work on an aircraft.[1]

The matrix organization rationalized the flow of technical information within the firm and rationalized the accumulation of resources and capabilities in "an age of massive engineering." The 1950s witnessed a strong movement toward national engineering unions. The vicissitudes of contract cycles caused regular lay-offs and constant circulation of engineers between firms. The engineering unions unionized portions of McDonnell's professional staff and agitated for nationally-defined job descriptions and work conditions, as well as for pension and vacation funds vested by the federal government. In response, McDonnell structured its departments to function similarly to a union within the firm. The departments were to provide a center for secure, stable, career advancement. Each individual engineer was evaluated by two bosses: one in research, one in development. His time was owned by the functional department, but he billed his time to the projects. In the project office, he found the excitement of a crash team effort, a connection to national security goals, and plenty of money for the application of ideas. When the program ended, the engineer went back onto departmental overhead, where he updated his technical expertise and analyzed what he had learned in solving a technical problem for the project. If he was ever fired, his departmental title explicitly defined his epistemological function, and he could point to the aircraft models on his trophy shelf which substantiated the utility of his expertise.

It is not known who first used the term "matrix organization," nor which firms pioneered its use. However, by the early 1960s every aerospace firm had adopted a type of matrix organization, and every engineer with whom I have spoken could explain how their career was shaped within a matrix. Still, matrix organization as a concept was not vital to transformation of the epistemology of aircraft design. Rather, the concept of matrix organization served to clarify the social context of aircraft construction. That is, a matrix organization chart "represented" the contemporary quest to prove some mutual symbiosis between basic research (as defined by Vannevar Bush's *Science: The Endless Frontier*, a plea for government funding of university science) and large-scale, technocratic projects (as defined, post-hoc, by Walter McDougall in *The Heavens and the Earth*). Many other types of firms--especially in those industries that billed engineering hours to contracts--invoked the image

---

[1] Organizational sociologists like James March, Herbert Simon, and Chris Argyris focused their work in the 1950s on similar questions of how bureaucracies controlled the decision-making processes of their members by dividing labor to control the paths along which they search for solutions to problems. This theoretical framework suffused the concerns of contemporary managers.

of matrix organization to clarify how they linked basic research to product development.[2]

For instance, program managers in the software industry combined existing code packages, then tailored them into one complete software program to suit the needs of a client. The Systems Development Corporation of Santa Barbara in 1957 received a $20 million contract from the Air Defense Command for the command and control program of the SAGE air defense network. Over the next five years, SDC hired and trained 7000 computer programmers; SDC managers claimed, "We trained the [programming] industry" [1]. SDC made SAGE a forerunner of the modular construction of software. Programmers based at the Santa Monica headquarters organized themselves into skill centers--requirements, analysis, design, testing--each of which turned out trained personnel and discrete code packages. (These packages were later refined into JOVIAL, a higher order language like FORTRAN and COBOL that described code packages in plain English.) Field programmers, acting as program managers, then selected the code packages required for one of twenty-six SAGE sites and mixed and matched the appropriate codes to the specific geography and weapons. In February 1958, SDC adopted a more formal matrix organization by giving the program teams visibility equal to that of the skill centers. SDC's organization chart had field programmers report to Model Mangers who supervised a general software configuration. These model managers then reported to the Program Managers for SAGE software, SAGE training, or the Air Force command and control system. In sum, SDC strengthened its program hierarchy to better utilize its functional departments.

Bechtel Incorporated, a civil construction firm based in San Francisco, moved toward matrix organization in the mid-1960s by strengthening their functional departments. The fixed-cost, turn-key contracts Bechtel had depended on in the past (and which had led to a project-oriented line and staff structure) had assumed an antagonistic relationship with their clients. Furthermore, Bechtel projects became more complex, making previous methods of estimating time and cost inadequate. Bechtel hoped to receive more cost-plus contracts for serving as architect-engineers on programs, much like the systems engineering and technical direction contracts sought by Ramo-Wooldridge in missile programs. Before Bechtel could bill a client for such engineering services, Bechtel would have to better define which engineering services they were providing. As one company report later stated, program management "was logistically similar to eating an elephant: we had

---

[2] Law firms typified the billing-oriented firm, like other professional services firms, consultancies, accountancies, hospitals. Hollywood is run by independent producers who cull their Rolodexes looking for specialists to complete a project. Custom software, mainframe computers, telecommunication services all depend upon billable hours. By the 1970s any high-tech firm attempting to launch a new product would name a program manager to marshal it through the firm and simplify cost accounting, even for those products that would be sold at a set price in a mass market.

to first carve it into chunks that were easy to digest" [4]. Bechtel started assigning a manager to every "controllable function," that is, any engineer whose work could be considered a billable activity was assigned to a department manager. The more people department managers captured into departments for engineering, procurement, or accounting, the fewer people fell under the umbrella of general Bechtel corporate services, thus decreasing the percentage Bechtel charged as general overhead on their contracts. Functional departments, rather than a central estimating bureau, now calculated all project estimates. Departmental managers had to prove their functional expertise was a discrete and necessary commodity and had to bill the projects for the manhours needed to finish a new plant, building or airport. Thus, in contrast to McDonnell and SOC, Bechtel strengthened their functional departments in order to create a balanced matrix.

The matrix organization chart represented the managerial skeleton of a large technological system. New groups of people hoping to attach some new component to a system--to an aircraft, an airport, a software program--had to attach themselves first to its matrix, that is, to its system of technical management. In short, the matrix depicted the connections between the many wide-flung social groups interested in building a technological system. The connections between the physical parts of this technological system were depicted, across an expanse of time, by time network charts.

Time Network Charts

The bar chart was devised by Henry L. Gantt and Frederick W. Taylor in the 1900s to help managers maximize their output of physical work by dividing it into more specialized labor. Into the 1950s, the bar graph remained the most widely used technique for graphically correlating work by time. Managers in high-tech industries often used the bar chart to portray intellectual labor because it depicted strict deadlines for the delivery of parts, and depicted tasks as running concurrently, rather than sequentially.

General Bernard Schreiver, the Air Force program manager for the Atlas ballistic missile, used a bar chart to make concurrency a centerpiece of his program. Rather than delaying the design of one part (say, the exhaust nozzle) until his engineers completed the design of a connecting part (say, the mixer/burner), Schreiver had his engineers design both simultaneously. This forced his systems engineers (Ramo-Wooldridge) to carefully coordinate the system interfaces. Engineering responsibility was thus decentralized, enabling subcontractors for component parts to make design decisions within interface constraints. Concurrency also forced Schreiver into massive redundancy, such as designing and fabricating two alternative nozzles in case one design failed. By contrast, program managers with more limited funds could not afford the luxury of buying redundancy as an antidote to concurrency.

Other flaws in the bar chart limited its spread through the newly programmed aerospace industry. First, if a client accelerated the deadline for a project, the program manager, in recalculating his chart, had to cut the time length of each bar and expedite every engineering task, even though most of the activities did not need to be rushed. Second, the bar chart did not show which engineering tasks had the most uncertain schedule, and which needed special attention. Third, the bar chart did not depict which tasks had to be done sequentially, perhaps because a part was so novel that system interfaces could not be fixed until prototypes were tested. Though useful to program managers, the bar chart did not fully represent how they envisioned their jobs. But the time network chart did.

The best example of a time network chart was PERT (Program Evaluation and Review Technique). It was designed, in the first half of 1958, by Gordon Pehrson of the SPO (the Navy Special Projects Office) responsible for the Polaris ballistic missile, with help from Lockheed, responsible for assembling the missile, and Booz, Allen and Hamilton, which provided the SPO with general computer services. The Polaris program was already well underway by October 1958, when the SPO imposed the PERT chart on the 3000 involved contractors and government agencies. Therefore, the SPO did not use PERT as it was later advertised, as a planning protocol, as a means of dividing and parceling out engineering tasks to make sure the project was finished on the tightest schedule. Still, PERT helped the Polaris program manager, Admiral William Raborn, pressure his contractors into keeping to the planned schedule.

The basic building block of PERT was an arrow marking the start and finish of an engineering task. But rather than simply layering the arrows as in a bar graph, PERT connected them into a network of tasks, each dependent upon another. Thus, to get from plans to a finished missile, the arrows showed the project splitting into subsystem tasks (the navigation electronics, aerodynamics, solid propellants) with these arrows subdividing into smaller tasks (the chemical composition of the propellant, methods of casting, testing specific impulse). Some of these arrows depicted tasks running concurrently, some sequentially. All of the arrows eventually depicted prototypes converging into subsystems, undergoing tests, then converging into the final system. Thus, if done in sufficient detail (only 5000 tasks were PERTed for the Polaris), the PERT chart showed how the daily tasks of every engineer contributed to the finished missile.

More importantly, the PERT chart showed how the finished missile depended upon every engineer finishing his task on time. That is, through the network of arrows ran one "critical path." This path showed those tasks that would take the longest to complete, which marked the shortest possible time for the completion of the whole missile. Raborn and Pehrson attempted to subdivide the tasks for each subsystem so that the path for each was equal in time. PERT depicted time as probablistic, allowing the SPO to calculate optimistic, likely, and pessimistic times for products that had never been built. However, if one task fell behind schedule, the path on which it lay then became the critical path. The task was marked with a red square, and the path on which it lay was marked in bold ink.

The PERT chart also came with a task force room, referred to by one consultancy as a "decision environment room." Every Monday morning Raborn assembled the program managers for each subsystem to review their progress against time. The manager of the subsystem on the critical path was intensely grilled. Often this manager argued that he needed more engineering overtime; often Raborn simply told him to ship his prototype to the next stage. Though PERT pointed out problems within the vast networks of engineers working on a project, management control was still done the old-fashioned way--haranguing.

Still, this new breed of program managers believed that PERT helped them solve a real problem: how do they learn to manage work they only do one time?[3] Though each weapon system was different, program managers learned how to represent each on a PERT chart, and thus learned to anticipate where integration problems might occur. The PERT chart served as a road map through uncertain terrain, yet various consultancies struggled to define which map symbology they should use. One author estimated forty variations on the network chart in 1964 [2].[4] Contractors complained about having to learn different reporting systems--and pay management consultancies to teach them--so in 1962 a PERT Coordinating Group, with representatives from each of the services and NASA, worked to standardize the network charts.

The Group agreed on the format and algorithms used in the network chart, but disagreed over what data they should require from their contractors. By PERTing a project, a military officer could demand interim reports on a contractor's technical progress as specific as the interim cost data they normally collected. More importantly, the Group insisted on cost-weighting to the time figures, and the result was called PERT-Cost (they renamed the original SPO charting PERT-Time). PERT-Cost combined the features of the PERT chart, which calculated the probability of hitting a schedule, with an industry-invented method which reported the costs of failure to keep up with the schedule, called CPM.

CPM was begun early in 1957, about the same time that the Navy SPO started working on PERT, by managers of the Dupont Company plant in Newark, Delaware. Dupont owned an under-utilized UNIVAC 1105 computer, had constant difficulty in scheduling plant construction, and housed a "systems engineering development group" trained in operations research. Operations

---

[3] Systems engineering, as a discipline if not an activity, served the same function. Systems engineering textbooks from the 1950s presented generic models listing each step in planning a new generic, high-tech project. System engineering theorists attempted to skeleton the epistemological process of planning, designing and testing new systems. In fact, the aerospace industry already used a generic method of building new weapon systems that revolved around the process of rewriting standards and specifications. But standards writing was the province of the functional departments, not the program managers.

[4] The Air Force translated PERT into PEP (Program Evaluation Procedure); the Navy Bureau of Yards and Docks used CPM (Critical Path Method); Dupont and C-E-I-R, Inc. offered PERT/RAMPS (Resource Allocation and Multi-Project Scheduling); and several prime contractors, most notoriously General Electric, the Federal Electric Corporation of ITT, and Lockheed, imposed PERT variations on their subcontractors.

research, as a discipline, focused on flow problems--distributing goods to retailers, planning highway and airport use, covering the sky with a limited number of radar units, or blanketing the Soviet Union with a limited number of missiles. Because flow problems also paced the construction and maintenance of chemical plants, Dupont engineers adopted an operations research method called parametric linear programming and built computer models of the process. They first called it the Kelley-Walker model, after the Dupont and Remington-Rand scientists who invented it, and later PPSS (the Project Planning and Scheduling System). After applying the model successfully to schedule the replacement of a pipe at a neoprene plant in Louisville, Dupont decided to sell the program as the Critical Path Method.

In 1958 the team that devised CPM left Dupont and formed a consultancy, MacAuchly Associates, Inc. of Ambler, Pennsylvania, which cooperated with another consultancy, ENTELEK, Inc. of Newburyport, Massachusetts to teach CPM in corporate seminars. The major computer firms--Remington Rand, General Electric, IBM--quickly sold standard code packages that solved various CPM problems, and these code packages were crafted into proprietary versions of CPM by companies such as Lockheed, Aerojet General, Bechtel Incorporated, and Kaiser Engineering. Other consultancies, realizing that many subcontractors did not yet own a computer, taught non-computer means of calculating the CPM.

CPM utilized the same basic approach as PERT. Kelley claimed that a fundamental trait of all projects was that although "all the activities involved must be performed in some well defined order, little has been done to make explicit use of that fact" [3]. While PERT operated in a time economy (what varied was the probability of hitting a completion time), CPM operated in a cost economy (time was fixed and what varied was the probability of cost and schedule matching). That is, CPM assumed a certain number of manhours, at a certain price, to complete a task. If the task was changed, the schedule lengthened and the costs were likely to run over the budget.

CPM demonstrated that time affected direct and indirect costs differently. If firms took more time, and arranged work crews and equipment to obtain the least direct cost, they increased their indirect costs for interest payments and opportunity costs. Furthermore, customers often imposed bonus-penalty clauses into contracts to cover their business lost due to plant delays. Accelerating work in order to minimize indirect costs, however, resulted in overtime and shift work, oversized crews with less experience, and larger equipment. To calculate these trade-offs between direct and indirect costs, program managers could run a CPM chart.

CPM also calculated cost overruns. If a contractor reported that his part of a project was on budget, but behind schedule, then he would have paid for planned work that had not been done, and could then expect an overrun. For this reason, CPM weighted each scheduled task according to its expected cost, so that the program manager could track the progress of the budget and the hardware simultaneously. Budget and hardware were graphed on an S-curve, which, was a standard way of reporting cost commitments in construction projects. The costs curve was relatively flat at the start and finish of a project, with a surge in the middle. If a contractor reported a delay on a heavily cost-weighted item, then CPM predicted the resulting cost overrun. If the delayed item preceded work on another part in the Critical Path, then CPM predicted relative over-run in costs because all the contractors involved at subsequent states would have to make up time. Thus, while CPM gave some indication as to problems in financial control solvable only by standard accounting and auditing methods, it was more useful in predicting financial costs of engineering troubles.

Paul Hardeman, Inc. of Stanton, California did $51 million of work on the 1961 missile silos and test stands for the Air Force at Cape Canaveral. Because of the pervasive concurrency in missile programs, Hardeman, Inc. had to change the design of the base every time the design of the missile changed. Every time the Air Force submitted a change order for the base, Hardeman, Inc. used a CPM package to compute the ramified costs of delays over the entire project. They submitted the cost calculation to the Air Force; a federal court subsequently ruled that the calculations were valid. No company could legally make a profit on compensation for change notices, but many went bankrupt due to incorrect estimation of revised costs.

Construction firms almost universally adopted CPM, or some variant, by the mid 1960s. The construction industry accounted for 10% of American GNP in 1964, and the largest "custom engineering construction firms"--like Flour, Parsons Engineering, Kaiser, Brown & Root, and Bechtel Inc.--were growing substantially larger. Like the aerospace firms, these construction firms designed and built complex, one-of-a-kind technological systems: bridges and dams, factories and refineries, airports and submarine bases, and nuclear power plants. This work required that the firms find new technologies, define work packages for subcontractors, incorporate them into complex programs, and test the results. CPM showed their progress and, more importantly, showed the subcontractors how the flow of their work to the program determined its ultimate profitability.

The Business Structures of Technical Transactions

The study of program management helps historians understand the interweaving of financial and technical forms of organization; the notion of transaction costs provides analytical traction for historians studying program management. Alfred Chandler implicitly uses a notion of transactions costs in explaining why large corporations emerged in the late 1800s. That is, new processing and distribution technologies that allowed economies of scale and speed made new corporations dependent upon regular input of resources. Resources bought in an open market, where transactions were characterized by antagonism between buyer and seller, resulted in information and financial costs that managers could eliminate only through mergers. Corporations merged, horizontally and vertically, to bring in-house functions considered too risky to leave to the marketplace. Corporation managers arranged salaried specialists into line and staff structures to keep these systems functioning smoothly. However, Chandler writes little about how managers, once they brought in-house these functional units, continued to redefine the technological parameters of their systems of production. That is, even though the technological system was contained within the financial system of the corporation, corporate manages could continue to refine the technological system to further reduce the costs of incomplete technical information.

Thomas Hughes explains how some systems builders focused on cybernetic control mechanisms--Sperry's gyroscope, Westinghouse's transformers, Insull's load manager--that helped all the parts of a technological system transact more smoothly and thus obtain better returns on the capital invested in them. However, while his core concept of technological momentum explains the quantitative growth in institutions dedicated to the technological system, it does less to explain qualitative changes in organization which are so central to Chandler's work.

These two tools of program management--matrix organization and network charts--reflect how contemporary managers simultaneously tried to shape some concordance between financial and epistemological structures.[5] That is, program managers invoked the matrix to explicitly represent *which* engineering groups participated in building a new technological system and where their loyalties and competencies lay. The matrix depicts the "seamless web" of knowledge flowing between science and technology; it also depicts the balance between the accumulation and application of organizational capabilities. Program managers then invoked a network chart to explicitly represent *when* these engineering groups interacted to design a technological system.

While these two tools showed *which* engineering groups interacted and *when*, they did not show *how* engineering groups interacted. Nonetheless, because two engineering groups were incorporated under the same organizational umbrella, transaction costs between them did not automatically disappear. This integration only removed, as principle-agency economics tells us, the incentive for lying about prices, i.e., information asymmetries. However, once a group was listed on the matrix and plugged into the network

---

[5] While the matrix and network charts usually depict activities within the firm, both can show how groups were tied to a technical system that transcends firm borders. That is, if an aircraft manufacturer buys time from a university-run wind tunnel, the group that operates the tunnel can be simply appended to the matrix chart and their test data entered as an task on the PERT chart. Furthermore, the matrix can depict many levels of program work: from the daily work of an engineer (whether he reports to a project or department overhead), to the meshing of departments and programs within a firm (McDonnell), the relation of national programs to the industrial mobilization base (Atlas and Polaris), and the overlapping of national champions and multinational program consortia (Airbus and Tornado). The functional departments are always the warp of the matrix, the program groups are the woof. Both are woven into a net that catches, and constrains, all the institutions of the aerospace industry. Furthermore, the matrix helps us understand the structure of institutions not built around capital relations, and business-government relations when t[5]he government is the buyer. The "corporation," as an organizational form, belongs to capital (most national laws say corporations must try to make a profit, not simply make products); and the "complex," like the military-industrial one, belongs to the nation state (only the modern nation state can generate excess militarism and bestow excess profits). For all the people working to design a new machine, the corporation structure explains their financial relations; the complex explains their political relations; but the matrix explains their technical relations.

chart, the program manager knew that he must create routines for the smooth processing of technical ideas: such as upgrading intellectual property rights, and proposing appropriate engineering theory and standards for drawings, production and testing. In short, the representational structures of program management ultimately had epistemological goals. Any matrix organization possessed a natural system of checks and balances because constituencies for solutions to design problems regularly questioned the work of other constituencies. Any network chart reflected the assumption that no component group could design a part without appreciating how the entire system was affected. Matrix organization and network charts thus helped rationalize the throughput of ideas. Management charts represented the epistemologies of systems builders and structured the flow of information and authority through an organization in order to replace, or mirror, their epistemological methods.

References

1. Claude Baum, *The System Builders: The Story of SDC* (Santa Monica, 1981) 43.
2. Arch R. Dooley, "Interpretations of PERT," *Harvard Business Review* (March-April 1964) 160-172.
3. J.E. Kelley, Jr., "Critical-Path Planning and Scheduling: Mathematical Basis," *Operations Research*, 9 (May 1961) 296.
4. Syncrude Canada Limited, Mildred Lake Project, Canadian Bechtel Limited Job No. 9776, *Project Management: Historical* (Bechtel Corporation, 1978) 15.