Two hat management—Project management with a difference

by ROGER W. KLEFFMAN
United Air Lines
Englewood, Colorado

INTRODUCTION

The on-line systems of the sixties were marked by an exuberant and dynamic environment of continual system implementations. New hardware, new operating systems, application growth outstripping system capacity—all led to large projects with dedicated manpower and hardware resources trying to achieve superhuman development schedules. The seventies are a different environment. New hardware and operating systems are introduced, providing increased capacity not only for application growth, but also for implementing new applications within the system.

The new environment was present at United Air Lines after cutover of an on-line airline reservation system in early 1971. The software was stable, and there was excess system capacity for new applications. All system and application program segments, on-line and off-line, were designated as specific responsibilities of first line program managers, who in turn managed programmer/analysts. There were no additional programming resources in any staff areas—the line had responsibility and accountability for all programming work effort.

Within this organization structure, three types of work would be done; correction of software errors, modification to existing coded functions, and major enhancements or development of new software products. The latter item would be carried out by means of projects. The projects would be of temporary duration; lasting only long enough to implement the new software. Upper management foresaw several problems:

- Dedicated resources for a project would be more costly than utilizing line talent on an ‘as needed’ basis.
- Illogical lines of management when programmers work for some period of time for a project manager, then revert back to the line manager for performance review.
- The need for technical personnel development as the old, reliable and talented people are committed to the choice projects, while new people are relegated to maintenance assignments.
- The requirement for line involvement in project planning, leading to line commitment to achieve project goals.
- The need for managerial development in maintenance and low software development activity areas.

This paper presents the solution United Air Lines developed to answer these problems; Two Hat Management, with a first line manager playing two different roles; a line manager controlling manpower resources and responsible for specific functions; and project leader, planning and coordinating the activity of several areas to achieve project goals.

Denver was the location where Two Hat Management was developed and implemented. This facility has a PARS system (programmed airline reservation system) under the ACP (Airline Control Program) operating system on a IBM 360/195. The current application and software systems include more than 500,000 instructions and more than 120 programmer/analysts maintaining and adding to the system by means of projects. Software projects generally exceed six calendar months and either add or modify 50,000 instructions annually. These conditions meet the criteria for large software system projects advocated by Aron.1

The paper is organized into four sections to develop and present the material. The first section introduces the environment and scope of the paper. The second section presents the organization structure and functional line management responsibilities. The third section discusses the formal procedures used to implement Two Hat Management in this organization structure. The fourth section summarizes the results of Two Hat Management in the work environment of the last two years.

ORGANIZATION

This section briefly presents the organization structure encompassing Two Hat Management.

Data processing services are provided to many departments and divisions of UAL, Inc., the parent organization of United Air Lines. The divisional organization concept with related profit accountability is the primary method of organization. The Computer and Communication Services Division (CCS) is responsible for almost all data processing activity within UAL. Figure 1 presents the CCS Division organization. The division is headed by a President with a very small supporting staff. There are four line functions reporting to the president: communications and technical services, computer service coordination, resource center and computer services. The latter two include almost all the division personnel and
resources. Appendix A presents a more detailed description of the organization management and functional responsibilities. The resource center organization is responsible for the computing hardware and system software at four computer sites. Consolidation of resources into a single computer center supported by remote sites is the long-range aim of the resource center, and is reflected in its organization. The computer services organization performs application programming for internal UAL users by means of five application programming centers. A summary of the location, equipment and applications is presented in Table I.

The application centers are in close proximity to the user groups, thus promoting decentralized development, while achieving the benefits of centralized resources, though along somewhat different lines than those described by Tomaszewski.

This organization is characterized by strong first and second line management, accountability for results, with check-and-balance functions incorporated in upper management levels. This is the contrast to check and balance by separation of responsibilities advocated by Smith. The most critical areas are those of interfacing and coordinating. There are three levels of interface and coordination: between first line functions within a department (e.g., first line functions in the resource center), among major departments (e.g., the resource center, the communications and technical services and all computer services' departments), and the CCS division with other UAL divisions (e.g., representing CCS with Flight Operations). This is a responsibility of the Project Support group, which provides project coordination and training services to the line organization. Without strong lines of communication, this particular organizational structure is vulnerable to project discontinuity resulting from lack of coordination. The means of accomplishing this coordination through formal procedures is the major element of this paper.

PROJECT IMPLEMENTATION PROCEDURE

This section presents the procedure used to implement the Two Hat Management concept. It features the meshing of the management organization with checks and balances established by formal procedure.

First line management and work authorization

As emphasized previously, the first line manager has functional responsibility for designated software. Only the first line manager can direct work on the software. The work environment has three types of work; with a reporting system to track progress.

1. Error Correction—defined by a discrepancy between the expected result (defined by documentation) and the actual result. An FPL (functional problem log) is originated to authorize work.

2. Modification to existing functions—representing an extension of a current function. A MER (modification...
evaluation report) is originated to identify work effort and track implementation, if authorized.

(3) Projects or large modifications—usually a new function to be added to the system. Projects are assigned unique account numbers, and tasks identify work units.

**Project management and implementation cycle**

The project leader does not control resources or authorize work, he is responsible for managing the project: planning, scheduling, coordinating and reporting progress. Deficiencies and problems are reported to his immediate manager (second line manager), and resolutions are obtained from control group decisions.

The control group is composed of second level and higher department management. The control group reviews project status on a weekly basis, and is responsible for short- and long-term departmental planning and objectives.

The project follows a cycle of definition, planning, implementation and maintenance as illustrated by Figure 2. A Project Workbook is originated and is the repository of all project information.

(1) **Project Definition**

A project leader is authorized by the control group. Assisted by a project coordinator from the Project Support Group, the project objectives and scope are prepared for control group review and approval. Objectives are the specific goals that the project will achieve, and are used to measure results. The scope is primarily a definition of planning effort required to size the project effort, a plan for producing the project plan. Included in the scope are:

- General statement of project definition.

**TABLE I—Resource Center and Application Summary**

<table>
<thead>
<tr>
<th>Location</th>
<th>Equipment</th>
<th>Operating System</th>
<th>Type of System</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chicago</td>
<td>UNIVAC 1108 EXEC 8</td>
<td>On-line</td>
<td>Flight Operations</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBM 370/155 IBM OS MVT</td>
<td>On-line, Batch</td>
<td>Finance</td>
<td></td>
</tr>
<tr>
<td>Denver</td>
<td>IBM 360/195 IBM ACP*</td>
<td>On-line</td>
<td>Marketing, Food Services, Hotels</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBM 360/195 IBM OS MVT</td>
<td>On-line, Batch</td>
<td>Time-sharing Option (TSO), Remote Job Entry (RJE) for internal UAL users and external customers</td>
<td></td>
</tr>
<tr>
<td>San francisco</td>
<td>IBM 360/65 IBM OS MVT</td>
<td>Batch</td>
<td>Maintenance Operations</td>
<td></td>
</tr>
</tbody>
</table>

* IBM ACP is the Airline Control Program.

(2) **Planning project**—A planning project is the preliminary step to a very large development and implementation projects. The planning scope of effort and resource requirement is sufficiently large to require a separate project whose products lead to identification and authorization of the main project.

(3) **Design project**—produces the design development and documentation necessary to support the implementation of a requested service.

(4) **Implementation project**—includes all work necessary to make the requested service available. Programming, testing, performance measurement, hardware installation: all are valid tasks within this project.

(5) **Maintenance projects**—a phase which includes maintenance and growth workloads and resource allocations. All workloads of this type are organized into three categories:

- System maintenance
- System growth
- System performance and measurement predictions

It is important to emphasize that planning, design and implementation projects are separated into individual projects based on the control group's estimate of the size, complexity and controls required to perform these phases of the development cycle. When appropriate, these three phases are combined into a single project.
(2) Detail Plan Development

A detailed specification of tasks, resources and schedule of the project is prepared by the project leader for review and approval of the control group. The project leader chairs planning meetings attended by the line managers and/or their technicians. These meetings review the project objectives and scope against the line functions to identify all the tasks and dependencies of the project. Each line area is responsible for identifying and specifying all tasks in their respective areas, and the resources needed to accomplish the task. A functional specification developed by the user is the basis of all planning and resource estimates in the detail plan. In turn, the detail plan provides the developmental and recurring costs which will be charged back to the user.

Planning development is iterative in nature. An overall network is produced first, with subsequent requirements added in. Investigative tasks may be undertaken in order to clearly identify project tasks. The output of the detail plan includes:

- Task definitions—produced by the line area which will have the responsibility for later accomplishment. The line involvement in the planning phase provides the means for strong line confidence and commitment for the project objectives. Appendix B presents the basic task document.
- Dependency network which identifies the sequence of task performance. A PERT chart is another name for this chart.
- Resource requirements for accomplishing tasks. Since these are generated by line areas which will accomplish them, there is no later distrust and disavowal of the estimates.
- A schedule of the task in terms of elapsed time and desired completion date to accomplish the project. The schedule reflects the timing as if there were no other project, with priority conflicts resolved later.
- Major project milestones, which are applied to network and schedule.

Specifically included in the schedule are several critical control points which furnish opportunities for technical review and thorough check-out of the application system product. These include:

- Initial Reviews—these are scheduled on an as needed basis. As a minimum there is a review or audit of the detail plan itself. The detail plan is presented to the entire line management for review and comment, including those areas seemingly not impacted by the project. Depending on scope and complexity of the project, additional reviews may be scheduled to review timing, events and dependencies associated with cutover.
- System Test—an integrated test of the coding under regulated and low volume conditions. Scripts which identify inputs and responses are generated by the line area responsible for the code.
- Pre-operational Test—a test of the procedures and actions required by the operations group. This milestone is an acceptance test by operations of the code. Without it, operations will not accept responsibility for production runs. The program documentation is the script, and conditions are created which exercise the documentation.
- Operations Test—a volume test of the code. Its function is to provide the widest possible exposure to the total environment. Again, scripts with inputs and responses are generated by the line area responsible for the code.
- Functional Demonstration—this is an option by the users to conduct a systematic test and check out of the code against the user procedures. The common denominator here is the user-generated functional specification. From it programming develops the programs which is the computer solution of the specification; while the users generate their procedure manual. This insures correspondence between code and user functions.
- Cutover—this is the loading of the code into the system and marks acceptance by the user and completion of the project. Hereafter, all work will be performed on the basis of FPL/MER rather than project.

The detail plan is presented to the control group, along with information relating to the project priority and schedule impact. The control group resolves known conflicts and authorizes the project leader to proceed.

The project leader reconvenes a meeting with the first line managers, identifies the project priority, and requests commitment by the line. The line managers factor the tasks into their work schedules and respond with a completion date for each task. This is the line commitment. If there are differences between the commitment date and desired schedule date which impact the project, the first recourse for resolution is the project priorities. Where this does not resolve the problem, the project leader refers it to the control group who will authorize either a higher task priority, or, accept a later project completion.

These task commitments then become measures of line performance as the project is implemented.

(3) Project Implementation

Project implementation is conducted in an environment where the project leader maintains and coordinates the project. He will report project progress to his immediate manager, but has commitment to the project objectives, irrespective of any line resources he may control. On the other hand, tasks are implemented under line organization control. Programmer/analysts report to their immediate supervisor, and their performance is measured and reviewed in this context. Programmers can work on several tasks and/or other work (MER, FPL).

Higher priority work received by the line manager can impact project work. The line manager reports progress and problems to his immediate manager in terms of task date and manpower commitments. The project leader reports in terms of project orientation and milestones, surfacing problems for control group resolution as required. Independently, the project coordinator furnishes a report of progress to his
immediate supervisor, as a check and balance. (A part of the function of the manager of Project Support is that of Devil's Advocate!)

Several major actions occur during this phase:

- Additional tasks may be required for a variety of reasons.
- Extended events may require new priorities—handled, as before, through the control group.
- Modifications to tasks may be required, created by users with the task amendment form (similar to the task description form).
- Reporting of task completion, accumulation of resource expenditures, and accumulation of products as defined in task descriptions.

As testing begins (systems, pre-ops, operations and functional demo), errors are tracked on special FPL reports and fixed on priority basis. Simultaneously, user requests for modifications are evaluated and either deferred until after cutover, or incorporated into the existing schedule and implemented, with schedule adjustments as necessary.

In terms of charge-back, the user will be charged the planned development cost, regardless of actual cost. This, too, becomes a measure of performance, both for the CCS Division, as well as the project leader.

(4) Project Maintenance

The maintenance phase is entered after cutover. Upon reaching this phase, no further work can be charged to the project. All work is either FPL or MER. At this point, the project leader produces a project report, which reviews the project in terms of schedule, resources used and objectives. There is also a critique of the project in terms of items which the project leader feels are important for subsequent projects. The critique can contain suggestions or advice, referencing procedural problems, recognition of work performed or general counsel in the area of leadership.

SUMMARY AND CONCLUSION

With longer-lived computer systems possessing sufficient capacity for the addition of new application software systems, project implementation enters a new phase. Systems continually grow larger, and the organization structure must be able to accommodate both normal system activity (bug fixing and enhancements) as well as development and implementation of new application systems. Implementing new projects in this environment must be done without interposing project organization between the line organization and the ongoing system. The management method described in this paper seeks to incorporate project task activity with the line responsibility for accomplishing the work.

The procedure has been successfully used at the Denver site for more than two years and a dozen completed projects (additional projects are in operation but not completed). The procedure works in the defined environment and has provided four tangible benefits. Management acceptance, enthusiastic endorsement and application of these benefits provide the substantiation for their results.

1. Better utilization of scarce programmer resources.

Programmers coding in assembly language for a real-time applications environment are scarce. Using the two hat method, first line managers schedule programmer/analysts for a variety of work, including project tasks, without dedicating programmers to projects. In this way, a programmer is busy with FPL (fixing errors) and MER (modifications and enhancements) work while also working on a six-man week project task spread over 12 weeks, for example. The manager will also adjust work schedules dynamically to include new, higher priority work.

2. Increased technical development of programmers.

Under this system, new people start on FPL work, then working on MERs and later project tasks. In projects, the progression includes modification to existing code and, later, coding of entirely new program segments. Participation in the various test phases provides additional valuable experience. Progression is not sequential; programmers are scheduled for several concurrent work efforts (FPL, MER, tasks) of comparable complexity.


The progressive series of tests described in Section 3, provide three to four levels of check-out of actual system quality against the specified requirements. These result in an application system which is stable and user accepted.

4. Conducive environment for management development.

Twelve out of fourteen first line managers (86 percent) currently are project leaders of significant projects. Of thirteen projects reported in on this paper, there were ten different project leaders. A first line manager always has tasks from other projects to implement, and sooner or later a first line manager will get one (or more) projects of his own to handle. There is no escape from the demanding requirements of project leadership and its attendant interaction with higher management and user levels, as well as with other first line managers. Practical situations abound for management development through experience.

Results

Thirteen projects undertaken between 1970-1973 form the basis of the results reported in this section.

- Work Mix
  
To put the results in perspective, and to emphasize that the environment is an on-going system, a typical work mix of more than 60 applications programming analysts includes:

Training and non-production time 16 percent

Modifications and error fixing 37 percent

Project development activity 47 percent

Currently, project development activity exceeds 70 percent of planned efforts.

Often the programmers were involved in concurrent types of work (both development and non-development work), and many existing program segments were subject to multiple
concurrent changes. (Modification to one part of the program while fixing an error in another part of the program).

- Project Descriptions

The thirteen projects included two pure study projects. The distribution of project sizes is presented in Table II. Note that the project size is measured by actual man days of work effort, excluding any time spent in overhead activity. In order to put the figures into equivalent people requirements, an increase of 15 percent for overhead activity should be supplied. This additional 15 percent includes categories such as training, sick time, vacation and related items.

Two projects were mainly in the resource center area, seven were in the application center area, and one was a special software development project.

The special project was both the study and implementation of a high level language compiler. During the project, the scope, objectives and resources were increased beyond the original plan. The project was later deferred indefinitely at a point where resource expenditures reached 50 percent of the original estimate and 72 percent of the revised estimate.

In the main, the projects were concerned with implementing new application functions in less than a calendar year, were of moderate size and required tasks from almost all line areas. One significant exception to this was a resource center project converting the system from a 360/65 to a 360/195 mainframe, with some significant hardware additions. This project was of such magnitude (requiring more than 17 man years of effort) that it significantly impacted the combined statistical results of all projects. Consequently the results are presented both with and without the impact of this project to avoid distortion and provide comparisons.

- Project Results

The results do not include the special compiler development project which was deferred prior to completion. The two study projects are also excluded because studies (following Parkinson's Law) seem to take all the time allocated for them.

Of the ten implementation projects; eight were on-time (80 percent):

- 4 were within budget
- 1 exceeded budget

One was 1.5 man years, two were almost 1 man year, and one was more than 4 man years of effort.

One was 4 percent more than a two-man year estimate, one took 42 percent more than the 165 man day estimate, and the other took 71 percent more than a 2 man year effort.

An average of 58 lines of source code per man day for 132 days, resulting in 7,800 lines of code, are typical of project results.

Two projects were late, both resource center projects. The major one (main frame replacement) took more than 17 man years versus eight man-year estimate.

Included in the main frame replacement project objectives were:

- replacement of an IBM 360/65 with a 360/195
- installation of a new communication system (front-end processor and software operating system)
- a new communications line control unit (telephone line patch panel)
- modified system software and record type (enlarged control block and modification to a system-sensitive application record)
- conversion of support system hardware (from DOS to OS)
- addition of two new application systems

While the resource overage was more than 200 percent (17 man years of effort versus 8 man years planned), this project was only 3 calendar weeks late. This is a testimonial to the intense involvement by line personnel, resulting from their participation in building the detail plan and commitment to achieving the project objectives. Major problems contributing to the overrun included shortage of skilled manpower, inefficient test vehicles, and lack of coordination and control involving complex, interrelated tasks.

The other late resource center project was a complete reorganization to the on-line files, including increasing the number of disks from 112 to 120. This project took slightly more than twice the estimated 81 man-days. The primary contribution to overrun resulted from extra testing requirements. The impact of test requirements from both projects provide some additional concurrence with the results reported by Boehm.

The main frame replacement project significantly impacted the reported results combining all projects. Table III presents the ratio of actual work efforts experienced on the projects to the planned effort resulting from the detail plan. The figures presented include the results from the resource center, the application center, and the combined result for both organizations. Data is summarized for two conditions: for all projects together, and excluding the results from the main frame replacement project because of its heavy weight.

To interpret these results, note that the resource center actually spent more than twice (2.3 = 230 percent) the

<table>
<thead>
<tr>
<th>TABLE III—Actual versus Planned Efforts by Center and Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Resource Center</td>
</tr>
<tr>
<td>Application Center</td>
</tr>
<tr>
<td>Total Organization</td>
</tr>
</tbody>
</table>
resources they had planned, including the results from all projects. Excluding the main frame replacement project, the resource center spent only 30 percent more effort than planned.

The results show that the resource center experienced the greater problems and posed greater potential impact on the system (larger actual experience to planned effort ratios), while the application center work contribution had the dominant effort on overall results (there was more application work effort than resource center).

In conclusion, Two Hat Management works in the specified environment, and the anticipated benefits were achieved. These results were achieved during a period of moderate development work, allowing time for orderly development and fine tuning of the procedures. We are now about to enter a period of heavy software development, which should provide a final test for this management method.

REFERENCES


ACKNOWLEDGMENTS

The author would like to recognize A. C. Edmunds of IBM and L. F. McGrane of UAL, as the fathers of the Two Hat Management concept.

APPENDIX A—UNITED AIR LINES COMPUTER AND COMMUNICATION SERVICES DIVISION ORGANIZATION AND FUNCTIONS

Division organization

The organization chart is as shown in Figure 1. The Computer and Communication Service Division is organized as a profit center with accountability to the parent organization. The internal structure is organized to facilitate the delegation of profit accountability along functional lines.

The division has minimal staff support. There is a controller, responsible for the internal division profit plan and charge-back system for all computer and communications services provided to internal and external customers. The other staff function is a special assistant to the president, providing technical expertise, internal consulting services, and external marketing. There are four line executives: the Director of Resource Centers, the Director of Computer Services, the Director of Communications and Technical Services, and the Manager of Computer Services Coordination who provides liaison between the CCS Division and other UAL organizations: the primary customers of the services.

The resource center is responsible for providing the central site computer resources: the hardware (CPU, memory, files, tape drives) and its operation, the system software (including communication programming), monitoring the use of resources and accumulating data for four computer sites at three locations (Chicago, Denver and San Francisco).

The Director of Computer Services is primarily responsible for providing application programming for internal customer services, by groups called Application Development Centers. Currently there are five Application Development Centers. Two in Chicago, one providing application programming for finance, the other supporting the flight operations functions. Two in Denver, one servicing the airline marketing applications, the other supporting Hotels and Foods Services systems. The last center is in San Francisco and services the aircraft maintenance group.

The Director of Communications and Technical Services has responsibility for the communication lines and interface with AT&T. This includes more than 40 high-speed (2400 baud, or higher) dedicated, leased lines for on-line terminals, an intercompany nationwide telephone network and the installation and servicing of remote site terminals and communications interface equipment.

Division control and policy is set by means of periodic (usually monthly) control meetings with the president and directors, supported and assisted, as necessary by the staff and lower level line management. The purpose of these meetings are for coordination of planning and policy, priority resolutions, review of major project plans and action responsibility for problems. This approach is mirrored in the line organization. Each director holds a weekly control meeting with his staff. The primary difference between the division and department meetings is one of detail. The departments monitor all projects status on a weekly basis, and authorize and review all expenditures of line manpower, with no exceptions.

Resource center organization

The long-range aim of this function is consolidation into a single resource center supported by remote, auxiliary sites as required. The organization structure reflects this approach.

There are second level managers (managers to whom first level managers report) at each of the current resource sites, with second level managers responsible for each remaining function. These are summarized below:

- Manager of Operations—responsible for the central site hardware operations. There are two first line managers, one responsible for the daily operations of the hardware and communications, the other responsible for the support functions of job submissions, scheduling, procedures and operator training.
• Manager of Systems Programming—responsible for all systems and communications programming. There are first line programming managers (managers to whom supervisors, programmer/analysts and technicians report) for each system—one for all OS systems, and for the ACP system, one for the UNIVAC system and one for the Communications Control system in the ACP area.

• Manager of System Support—this responsibility includes all the OS utilities, the ACP utilities, file maintenance, and system performance and measurement (capturing and data reduction of resource utilization), each with its respective first line manager.

• Manager of Operational Programming—the functions of coverage programming (responsibility for keeping the on-line systems up and operating, monitoring and testing new and modified software into the on-line system), data base entry (integrity of data in on-line files) and schedule change data entry (schedule change data for the airline reservations system). Coverage is headed by a first line manager, the other functions are currently led by supervisors.

• Manager of Quality Control—this responsibility includes two functions: the test system and the library. The test system controls and maintains the test tools used in program and system check-out, from initial program unit test in an off-line, single thread environment, up through a volume test of the entire applications and systems. All programs complete strict testing requirements through a series of increasing integrated testing.

• The library provides the control function for tracking the current state of the system. It insures correspondences between program code and documentation. No programs enter the system without authorization from this checkpoint. The library also is responsible for the preparation, editing, classification, recording and filing of system documentation and maintenance of the technical reference library.

• Manager of Project Support—This responsibility is present in both the resource center and the applications development center, but each is responsible to their respective director. By analogy, if the resource center is the bone, the application development center the muscle, then the project support function is the connective tissue tying everything together. Project support provides the interface between and among organization elements, project planning and monitoring service (the IBM Project Management System (PMS) is utilised), project coordination (assisting project leader by preparing project papers, planning, status, meetings and reports) and programmer training.

Application development center organization

This function provides all application programming for UAL user affinity groups. Each application development group is headed by a second level manager and several first line managers. First line managers are responsible for functionally related programs. All errors, modification or new programs are developed under the control of the first line manager. The programmer/analyst is accountable only to the first line manager, including all work performed on projects.

As with the resource center, there is a manager of project support, with similar responsibilities. In addition, there are project support managers reporting to the manager of application development, with the same responsibilities previously described. Here, the project support concept has been extended to a lower level to insure that needed functions and coordination will be accomplished.

APPENDIX B—PROJECT TASK DEFINITION FORM

This form describes the work to be performed as a part of the project. In order to assure that adequate planning has been accomplished, tasks must be defined in sufficient detail to allow its assignment and to establish its priority within the framework of established workload. The organizational philosophy requires that the task be defined such that it is assignable to a specific first line manager.

The task sheet has four basic parts:

1. Project identification and description—prepared by the project leader, assisted by the project coordinator with input from the first line manager.
2. Resource requirements—completed by the first line manager with his estimates of the workload.
3. Attachments—completed by project leader, used to include supporting material.
4. Task responsibility.

This form exists for the life of the project, first defining the work effort, later recording the actual work performed, and finally, providing a history of completed work and product in the Project Workbook.
COMPUTER SERVICES  UNITED AIR LINES

PASSENGER SERVICE SYSTEMS

FORM #50 R 12/16/71

TASK DEFINITION COVER SHEET

PREPARED BY:

PROJECT INDEX:________________________ DATE:________________________

PROJECT NAME:________________________________________________________

TASK INDEX:__________________________________________________________

TASK NAME:___________________________________________________________

TASK DESCRIPTION (ATTACH ADDITIONAL SHEETS AS REQUIRED)

RESOURCE REQUIREMENTS

1. TASK COMPLETION DATE DESIRED:________________________

2. RESOURCE ESTIMATES:  MAN DAYS____  MACHINE HOURS____

3. TASK COMPLETION COMMITMENT DATE: _______________________

4. ACTUAL TASK COMPLETION DATE: ___________________________

   RESOURCES USED:  MAN DAYS____  MACHINE HOURS____

ATTACHMENTS:

   _____ 1. ADDITIONAL TASK DESCRIPTION (CONTINUED FROM ABOVE)

   _____ 2. END PRODUCT LIST AND DESCRIPTION

   _____ 3. DEPENDENCY/CONSTRAINTS DESCRIPTION (DATES, OTHER TASKS, ETC.)

ASSIGNED TASK MANAGER________________________  PROJECT MANAGER______

SEE 8-00-28 FOR DESCRIPTION OF FORM