C

# Center for Advanced Computation

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
URBANA, ILLINOIS 61801



CAC Document Number 164
JTSA Document Number 5510

## Research in Network Data Management and Resource Sharing

## Research Plan

June 23, 1975

CAC Document 164
JTSA Document 5510

# Research in Network Data Management and Resource Sharing

Research Plan

Prepared for the
Joint Technical Support Activity
of the
Defense Communications Agency

under contract
DCA100-75-C-0021

Center for Advanced Computation
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

June 23, 1975

Approved for release: Peter A. Alsberg
Peter A. Alsberg, Principal Investigator

Table of Contents

# Introduction

## Background

A three year research plan is presented. The purpose of the research program is to develop distributed data management and resource sharing technology for application in the World Wide Military Command and Control System (WWMCCS) Intercomputer Network (WIN). This work is supported by the Joint Technical Support Activity of the Defense Communications Agency.

As part of the preparation of the research plan, the state of the art in network data management and related technology was surveyed. In addition, WWMCCS application requirements were surveyed. Technology support requirements for WWMCCS applications and technology interdependencies were identified. Where appropriate, preliminary research activities were undertaken to expose problem areas and to estimate research difficulty and probability of success. This work is described in eight support documents (table 1). These documents form the basis for this research plan. Familiarity with the concepts of distributed data management and with the support documents is important to a thorough understanding of the research plan.

Much of the basic technology required to support WWMCCS networking does not exist. Frequently, there has been little or no research in critical areas. In this plan, twenty-four research areas are described. Needs, priorities, and interdependencies are identified. Fifteen of the areas are directly addressed as part of the three year Center for Advanced Computation (CAC) research program. The remaining nine areas are of too high a risk relative to their payoffs or, due to limited CAC resources or the existence of superior external expertise, are better addressed by other agencies. In these latter areas, close cooperation between research groups is necessary, since the technology areas are interdependent. Thus a modest CAC staff commitment to those research areas will be required.

| CAC Document Number | JTSA Document Number | Title | Date |
|---|---|---|---|
| 149 | 5502 | An Annotated Bibliography to Network Data Management and Related Literature | April 1975 |
| 150 | 5503 | A State-of-the-Art Report on Network Data Management and Related Technology | April 1975 |
| 151 | 5504 | WWMCCS System Summaries | April 1975 |
| 152 | 5505 | Survey Report | April 1975 |
| 159 | 5506 | Scenario Report | May 1975 |
| 160 | 5507 | Application Summary | May 1975 |
| 161 | 5508 | Technology Summary | May 1975 |
| 162 | 5509 | Preliminary Research Study Report | May 1975 |

Table 1

Support Documents

## Overview of the Plan

The research plan has four major components:

1. the analytic modeling of distributed data management systems,
2. the development of experimental distributed data management systems,
3. an optional component to develop intelligent terminal technology, and
4. a technology transfer component.

The fifteen research areas are addressed within these components. Each area requires a theoretical activity to propose new or upgrade existing technology. This is the analytic modeling component. Similarly, an experimental component is required to test the concepts proposed in the modeling activity and to provide empirical results and feedback for the refinement of the developing technology. Thus the modeling and experimental activities are tightly coupled and mutually supportive.

Since the areas addressed in this plan are extremely interdependent, they must frequently be simultaneously addressed during the research program and are expected to be readdressed throughout the program. Thus, assigning a detailed schedule of specific activities for each research area is inappropriate. Instead, points are identified where usable results are first expected for a given research area. It is implicit that the work will continue beyond these points in each of the areas.

The intelligent terminal research areas are separately analyzed so that they can be optionally excluded or included in the plan. If they are included, then the intelligent terminal research areas will be subsumed in the modeling and experimental components and will be tightly coupled to the other distributed data management and resource sharing research areas.

The technology transfer component addresses the problem of moving the networking technology into the WWMCCS community as it is developed. Within this component, technical assistance will be provided to JTSA and designated members of the WWMCCS community.

Deliverables

Technical progress reviews will be presented quarterly. It is anticipated that these reviews will require one or two days each. The CAC will describe the work to date, demonstrate capabilities and prepare in-depth tutorials and lectures where appropriate. JTSA will provide program guidance.

Technical reports, demonstrations, and technology transfer tasks will be delivered on an event driven basis as mutually agreed by JTSA and the CAC.

The goal of the three year program is to develop prototype network systems. The technology developed to implement these prototypes and the experience gained in their use should form the basis of functional specifications and design specifications for operational WWMCCS network systems. Four prototype systems are anticipated:

1. a manual distributed data management system in which the data base administrator is responsible for system tuning and the user interface will be rather low-level;

2. a self-tuned distributed data management system to investigate a variety of techniques for automated system tuning and to investigate higher level interfaces;
3. an automatic workload sharing system which will select tasks to be shared, farm them out, and coordinate their processing; and
4. an intelligent terminal with some local data management capabilities and user aids to support continuity of operation and improve the human interface to network application packages.

## Format of this Document

Four sections follow. They describe research needs, strategies, specific problem approaches and management considerations.

Research needs in network data management and resource sharing. Twenty-four different topics are presented. Each topic is assessed for its risk, its criticality to the WIN distributed data managment system, and its WWMCCS payoff if the research is successful.

Research strategy. The four components of the research plan are described.

Research plan. The plan of attack on each of the fifteen selected research areas is described. For each area, five aspects are considered:

1. purpose and scope,
2. application orientation,
3. state of the art,
4. proposed research approach (strategy and methodology), and
5. anticipated results.

Management summary. The schedule, personnel requirements, and deliverables are summarized.

## Major Research Needs

Table 2 summarizes twenty-four research areas. Detailed information on each area is contained in the support documents for this plan. The five middle columns in the table present an assessment of the risk that the research will not be successful, the criticality to a functioning WIN, the payoff of investing JTSA resources, the year of the plan in which usable results are expected, and the expectation that the major WWMCCS research requirements for that area will be satisfied in the three year program. Definitions of each term used in the table follow.

Risk is a measure of the probability of not producing results that are simple, usable and can be successfully transported to the WWMCCS network. An experienced and competent research group is assumed.

1. High risk means that difficult research is required in an area that is not well understood at present. Research paths that are likely to produce acceptable results are not known with any certainty.
2. Moderate risk means that difficult research is required in an area that is partially understood. Research paths that are likely to produce acceptable results are apparent.
3. Low risk means that the required research is straightforward. Appropriate research directions are known.

Criticality is an assessment of how critical an area is to the correct functioning of a reliable WIN with shared data bases and shared processing resources.

1. High criticality means that the area must be understood before any effective resource sharing can be realized on the WIN. A failure in this area would jeopardize major WIN goals and funtional requirements.
2. Moderate criticality means that the area must be developed to make network application packages usable by non-programmer personnel or is required for acceptable reliability or performance. A failure in this area may cause important performance, reliability, and usability goals to be missed. The network could still function but not up to specifications.

3. Low criticality means that the area is one in which advanced development is important for superior performance, reliability, or usability above and beyond minimum network requirements. A failure in this area means that the network would be successful but would fail to achieve its full potential.

Payoff is a measure of how additional research in an area (beyond the research already in progress - if any) will further the achievement of WWMCCS networking goals. (Survivability is not a goal at present. If it were made a goal, several entries in the following table would be upgraded to high payoff.)

1. High payoff means that it is essential to the correct functioning of the WIN that this area be well understood. Alternatively, this area has the potential for overwhelming improvements in cost or performance.
2. Moderate payoff means that it is important to the correct functioning of the WIN that this area be well done. However, incomplete solutions that will permit the WIN to function at moderate levels are acceptable. Alternatively, this area has the potential for significant cost or performance improvements.
3. Low payoff means that the technology is useful but non-essential to the correct functioning of the WIN. Alternatively, this is a valuable area, but one which is already adequately understood for WIN purposes or which is being separately pursued in a manner adequate for WIN purposes, and thus the expenditure of additional funds is not warranted in this area.

Expectation time for usable results is the number of years before the research activity can be expected to produce results usable in the PWIN or WIN. It is assumed that a research alternative to produce the results is selected from the available research plan options and manned at the suggested levels.

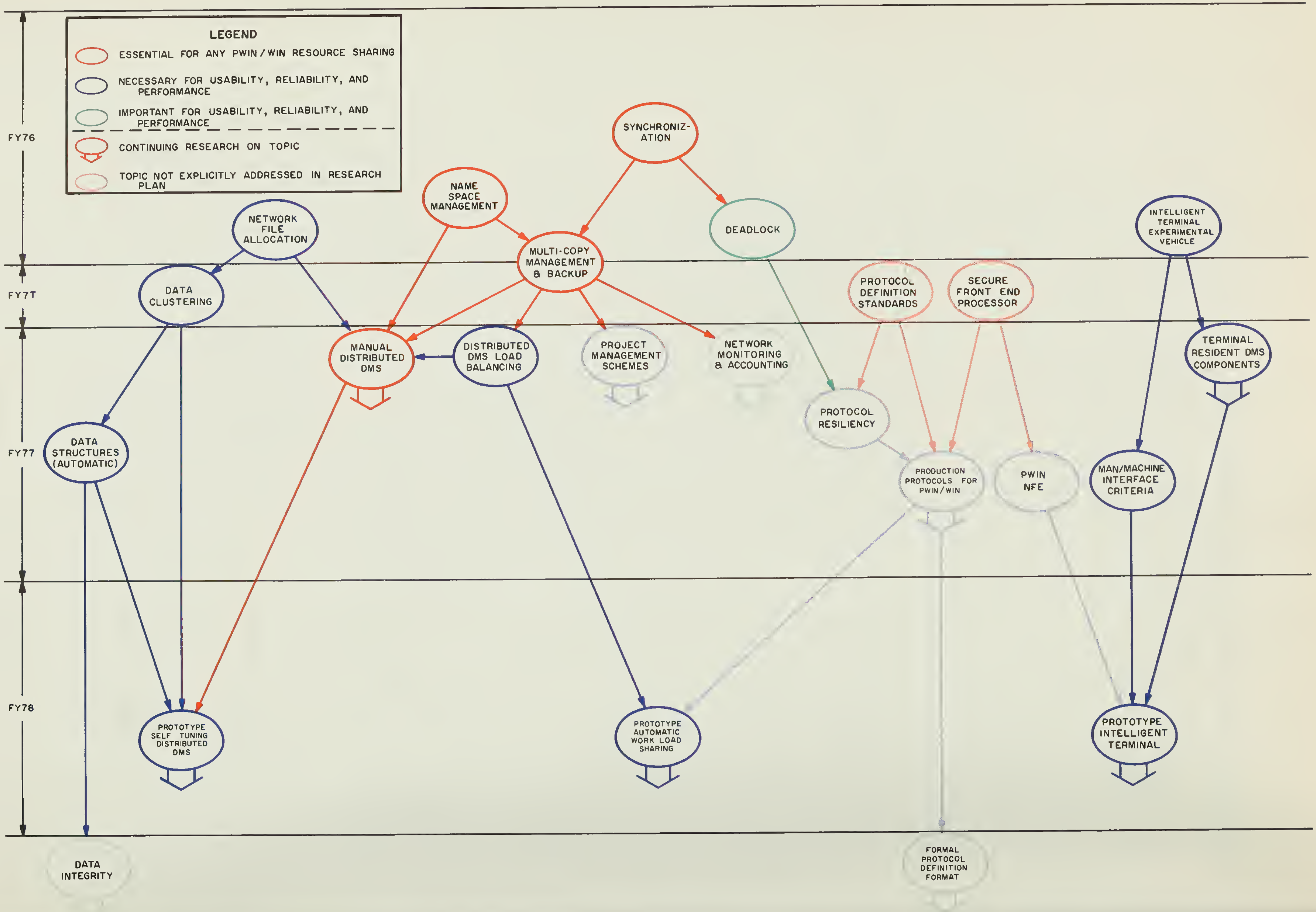
Research completion is an estimate of whether a relatively complete understanding of the technology area, for WWMCCS networking purposes, can be expected by the end of the three year research effort. Obviously, this is only an estimate based on our current understanding of each area. It is likely that a revision of this estimate will be made in some areas as a result of further research activities.

## Interim R&D Events and Critical Program Dependencies

Figure 1 indicates the time at which results are first anticipated to be available for each of the twenty-four research areas. The timing of these events is based on our current understanding of the problem and our best estimates. As the research program develops, it is expected that some events will tend to shift forward or backward in time. The product of the event may be a working paper, a technical report, or a demonstration of a concept, technique, or capability.

Criticality, as used in table 2, is shown by event color in figure 1. Those areas not addressed in this plan are shaded.

The areas in figure 1 are highly interdependent. Unfortunately, a figure showing important couplings, even if differently weighted lines are used to indicate degree of coupling, is so dense as to be unreadable. For that reason we have shown only the most critical dependencies, those where a logical time sequence exists or where significant results are required in one area before reasonable progress can be expected in another area. In some cases, a higher criticality area is dependent on a lower criticality area. This implies that the lower criticality area is not essential to the "dependent" area. However, if work is planned for the lower criticality area, it will tend to shape the dependent area and should precede the high criticality work. For example, the development of the manual distributed data management system (DDMS) has some dependence on the network file allocation work. Extensive file allocation work is not critically required for the DDMS. However, if extensive file allocation studies are planned, then it is appropriate to do them before the DDMS has evolved too far. Otherwise, the DDMS may not be able to exploit the results of a file allocation study.



LEGEND
ESSENTIAL FOR ANY PWIN / WIN RESOURCE SHARING
NECESSARY FOR USABILITY, RELIABILITY, AND PERFORMANCE
IMPORTANT FOR USABILITY, RELIABILITY, AND PERFORMANCE
CONTINUING RESEARCH ON TOPIC
TOPIC NOT EXPLICITLY ADDRESSED IN RESEARCH PLAN
FY76
FY7T
FY77
FY78
SYNCHRONIZ-ATION
NAME SPACE MANAGEMENT
NETWORK FILE ALLOCATION
MULTI-COPY MANAGEMENT & BACKUP
DEADLOCK
INTELLIGENT TERMINAL EXPERIMENTAL VEHICLE
DATA CLUSTERING
PROTOCOL DEFINITION STANDARDS
SECURE FRONT END PROCESSOR
MANUAL DISTRIBUTED DMS
DISTRIBUTED DMS LOAD BALANCING
PROJECT MANAGEMENT SCHEMES
NETWORK MONITORING & ACCOUNTING
TERMINAL RESIDENT DMS COMPONENTS
PROTOCOL RESILIENCY
DATA STRUCTURES (AUTOMATIC)
PRODUCTION PROTOCOLS FOR PWIN/WIN
PWIN NFE
MAN/MACHINE INTERFACE CRITERIA
PROTOTYPE SELF TUNING DISTRIBUTED DMS
PROTOTYPE AUTOMATIC WORK LOAD SHARING
PROTOTYPE INTELLIGENT TERMINAL
DATA INTEGRITY
FORMAL PROTOCOL DEFINITION FORMAT


FIGURE 1

| Technology/Techniques Needed | Risk | Criticality | Payoff | Expectation Time | Research Completion | Comments |
|---|---|---|---|---|---|---|
| Synchronization | Moderate | High | High | 1 | yes | Application of basic operating system and data base synchronization methods |

| Technology/Techniques Needed | Risk | Criticality | Payoff | Expectation Time | Research Completion | Comments |
|---|---|---|---|---|---|---|
| Synchronization | Moderate | High | High | 1 | yes | Application of basic operating system and data base synchronization methods to network environment |
| Name space management | Low | High | Moderate | 1 | no | Extension of name space to allow network wide object names and to allow naming of logical entities in a distributed data base |
| Network file allocation | High | Moderate | High | 1 | no | Placement of files in a network to satisfy cost, performance or non-technical goals |
| Multi-copy management and backup | Moderate | High | High | 1 | yes-basic | Design and implementation of algorithms for the management of a distributed data base, including update, query management and backup |
| Deadlock | Moderate | Low | Moderate | 1 | yes | Prevention and detection of deadlock in distributed data base systems |
| Intelligent terminal experimental vehicle | Low | Moderate | Moderate | 1 | yes | Basic test bed for future experiments with intelligent terminal technology |
| Data clustering | High | Moderate | High | 1+ | no | Techniques for grouping together data to improve performance, cost, security, etc. |
| Protocol definition standards | Moderate | High | High | not addressed | | Standards for the relatively unambiguous definition of protocols |
| Secure front-end processor | High | High | High | not addressed | | Design and implementation of verifiably secure front-end processor (ESD/MCIT) |
| Manual distributed data management system | Moderate | High | High | 2 | no | Implementation of an experimental distributed data management system |
| Distributed data management system load balancing | Moderate | Moderate | Moderate | 2 | no | Sharing of the workload in a distributed data management system to balance load on hosts |
| Project management aids | Low | Moderate | Moderate | not addressed | | Systems to aid in resource allocation and control |
| Network monitoring and accounting | Low | Low | Moderate | not addressed | | Measurement and analysis of network performance and use |
| Automatic data structure design | High | Moderate | High | 2 | no | Automated design of data structures to achieve desired cost and performance goals |
| Protocol resiliency | High | Moderate | Moderate | not addressed | | Design of Protocols which are tolerant of, resistant to and can recover from errors |
| Production protocols for PWIN/WIN | Moderate | Moderate | High | not addressed | | Development of protocols suitable for a production environment, i.e. resilient, easy to use, secure, etc. |
| PWIN network front-end | Moderate | Moderate | High | not addressed | | Development of prototype front-end for PWIN |
| Man/machine interface criteria | Moderate | Moderate | Moderate | 2 | no | Analysis of applications of the intelligent terminal as a user aid |
| Terminal resident DMS components | Moderate | Moderate | Moderate | 2 | yes | Development of high performance network DMS by moving some tasks to the terminal |
| Prototype self-tuning distributed data management system | High | Moderate | High | 3 | no | Distributed DMS with automated restructuring, load sharing, and file allocation |
| Prototype automatic workload sharing | Moderate | Moderate | Moderate | 3 | no | Automated system to balance host workloads by moving portions of the workload to other hosts |
| Prototype intelligent terminal | Moderate | Moderate | Moderate | 3 | no | System suitable for practical use |
| Data integrity | High | Low | Moderate | not addressed | | Formal definitons and algorithms for checking the consistency of data and correcting erroneous data |
| Formal definition of protocols | High | Low | Moderate | not addressed | | Formal protocol definition to aid in proving correctness and resiliency |

Table 2

Major Research Needs

## Research Strategy

### Overview

The research program has four components:

1. a research study component directed primarily at the development and analysis of analytic models for the design and optimization of distributed data management systems,
2. a concept test and demonstration component to illustrate the practicality of distributed data management and to explore those research questions which are primarily matters of implementation by constructing an experimental distributed data management system,
3. an optional intelligent terminal component to develop lower-cost/higher-performance data management systems and to improve the human interface to WWMCCS application packages, and
4. a technology transfer component to disseminate existing and new technology to the end-user WWMCCS community.

Within these four components, fifteen of the research areas in figure 1 are addressed. Details for each area follow in the research plan section.

### Motivation

The modeling and experimental system efforts are tightly coupled (see figure 2). The analytic modeling will allow the compact and effective evaluation of a wide variety of distributed data system questions and design alternatives. The experimental system will provide both a concept testing facility for the results of the modeling effort and a vehicle to demonstrate distributed data management functions. Further, the experimental system will provide feedback to the modeling component by identifying those aspects of the models which are incomplete or inadequate.

This two-pronged theoretical and experimental approach has been chosen for a variety of reasons.

1. It improves the utilization of professional research personnel. Independent effort may proceed on both the modeling and experimental systems with no great need for close synchronization between them.

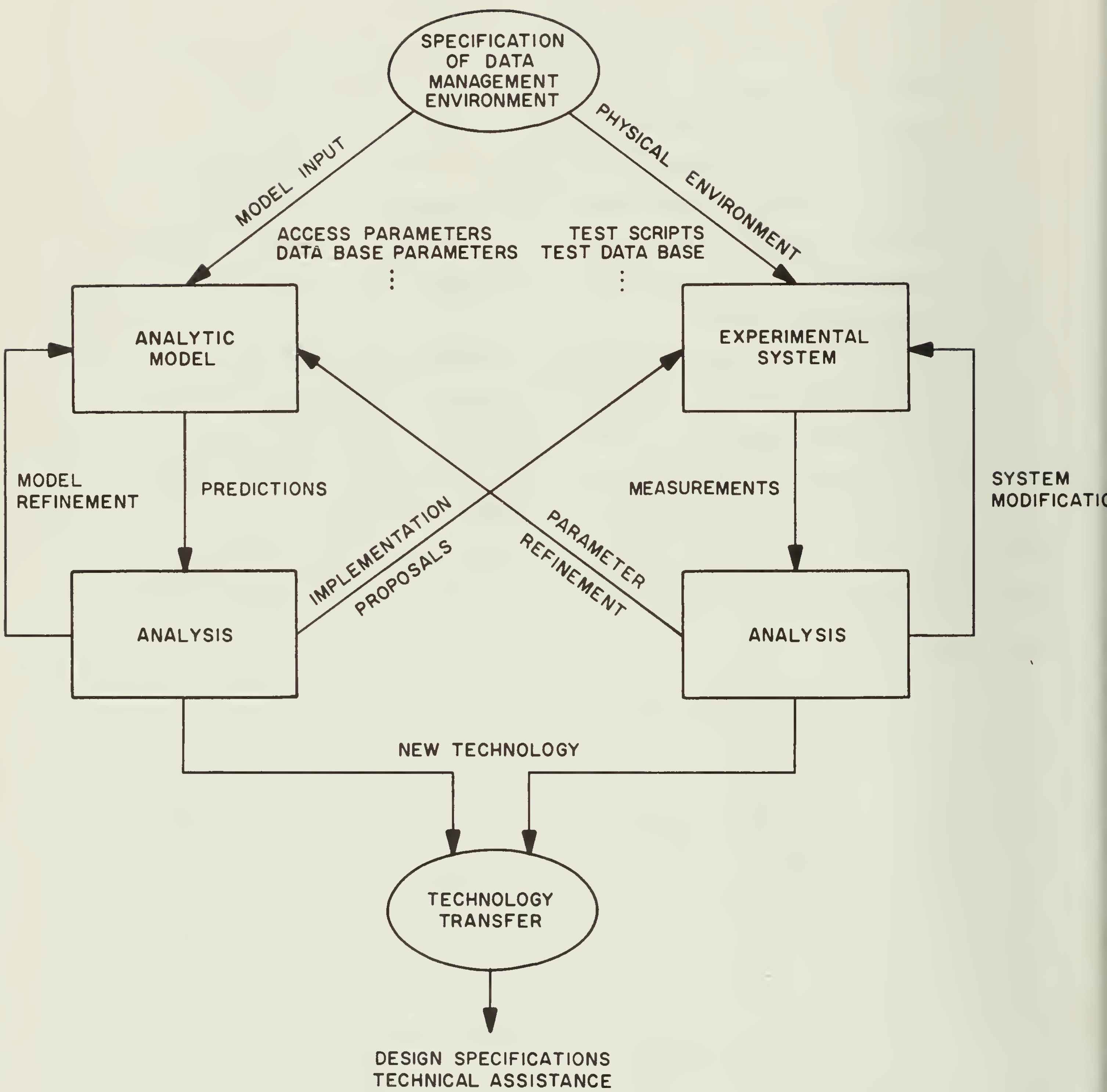



FIGURE 2

COUPLING AMONG THE RESEARCH PROGRAM COMPONENTS

2. The approach lends substantially more credence to the research results than can be obtained from either the modeling or the experimentation alone.
3. The approach provides a far more cost-effective attack on the questions of interest to and of impact upon the WWMCCS community than would occur by dedicating all the research manpower to a single, monolithic project, such as a large scale simulation or creation of semi-operational software.
4. We do not face the extremely expensive and difficult task of statistically verifying a large simulation because we will use a combined modeling and experimentation approach. The fact that an experimental system will exist means that actual experiments can serve to verify the modeling results.

The intelligent terminal represents the natural evolution of the concept of a distributed data base.

1. Preliminary experimentation has suggested that substantial overall system cost savings can be realized by providing each user with a powerful minicomputer in his terminal. This is due to the extremely attractive power/dollar ratio of mini-computers versus larger machines for some tasks (e.g. I/O). Where the application is I/O limited, as data management tends to be, terminal resident processing tends to improve both cost and performance.
2. Other benefits accrue in the area of human engineering. It is possible, for example, to present an application oriented interface to the non-programmer user and have the intelligent terminal perform the translation into a programmer oriented host (or network) control language.
3. The intelligent terminal, with its local data base capability, represents a highly survivable tool. Work can still be done if the host (or even the entire network) is unavailable.

The technology transfer component is one which is all too often neglected in research projects. The assumption is made that research results are so eagerly awaited by the eventual consumers that no serious effort need be expended in the dissemination of research products. This assumption is false. It is the responsibility of the professional researcher to make himself aware of the real-world problems and to assure that his results reach the consumers. We recognize our commitment to research dissemination and intend to pursue it actively.

1. As will be discussed in detail later, manpower will be devoted to direct technology transfer to the end-user WWMCCS community through a technical assistance program directed by JTSA.
2. We hope to enlarge the scope of the periodic contract progress reviews to include tutorial discussions on recent research results.
3. We intend to continue the timely publication of research results both in report form and in the open literature.

Analytic Model

The analytic model provides a means for the compact and efficient evaluation of basic design strategies and alternatives. The model will be highly parameterized and modular. The modular approach makes it possible to produce some early results using existing and estimated parameters. Later, as more sophisticated results are desired, better parameters can be obtained from simulation results; or from measurements performed on the experimental system, PWIN, or the ARPANET; or the parameters can be replaced with more detailed sub-models.

The basic classes of inputs to the model are:

1. parameters which describe the network behavior (e.g. bandwidth and transit times),
2. parameters which describe the intrinsic characteristics of the data bases, such as their size, their segmentation properties, and a stochastic characterization of access patterns (e.g., update/query ratios, expected hit densities in the physical data base, etc.),
3. parameters which describe the distribution policies to be used (e.g., whether or not multiple segment copies will be used or whether remote journalling will be done),
4. parameters which describe the load on the hosts and the network (e.g., the unused CPU or storage capacity at each host and network traffic loads),

5. parameters which are formal descriptions of non-technical constraints on the design of the distributed data base (e.g., mandatory locations of particular files),
6. parameters which describe the reliability of the components of the network (e.g., MTBF and MTTR for the hosts and probabilities of communication subnet failures that partition the network), and
7. parameters which quantify the costs of various network resources (e.g., secondary storage, CPU power, and communication traffic).

The major outputs of the model are:

1. an estimate of the overall cost of a given design alternative,
2. an estimate of the distributed data base system reliability and availability, and
3. an estimate of the system performance.

The model will consist of algebraic equations which relate the input parameters to the output parameters. This set of equations may ultimately turn out to be rather complicated. Therefore a major component of the modeling effort will be to provide guidance in the use and interpretation of the model.

Experimental System

It should be made clear at the outset that it is far beyond the scope of the planned research to produce an operational distributed data management system. Rather, we hope to produce a test-bed on which meaningful preliminary experimentation may be performed. Such a test-bed will also allow the exploration of those issues in which the theory is understood, but the practical details of implementation are lacking. The experimental system should be considered on an equal footing with the modeling effort.

Logically, the experimental system may be viewed as three components:

1. a local data manager,
2. a network data manager, and
3. a test generator and analyzer.

In practice, the distinction between these components will likely be blurred. The local data manager is much like any contemporary data base management system. It maps the user's logical actions into physical actions. The major difference is that the local data manager must have an interface to the network data manager. The network data manager is responsible for the creation and maintenance of a network virtual data base. The network virtual data base may not be physically resident on any one host. The principal function of the network data manager is to orchestrate requests to the local data managers who maintain portions of the network virtual data base. Because the system is intended to be a source of data for the modeling, a provision must be made for a systematic way of exercising the system and analyzing its performance. The test generator and analyzer performs this function.

The exact structure of the experimental system is a topic of ongoing research. Some guidelines have already emerged.

1. The system must be designed from the outset to be easily modified. This means, for example, that modularity becomes even more important than it is in more static systems.
2. The system must allow for added complexity as future research progresses. Examples of possible new features would include automated backup and recovery, and greater sophistication in network directory processing.
3. There should be extensive measurement capability which can be easily and selectively turned off and on.
4. The local data managers should allow the addition of data compression, data clustering and other potential research results.

Beyond these guidelines, a number of questions remain to be investigated. The most important of these is the nature of the local data managers. One alternative is to attempt to adapt an existing data management system to this role. The major difficulty with this approach is that adapting most data management systems would present a severe challenge. Their interfaces to the world are complex and cumbersome and generally not well suited to interfacing to another program. They also tend not to provide the necessary measurement tools. The major advantages to using an existing program are that a large amount of work has already been done, and we would not be faced with "re-inventing the wheel."

The other basic alternative is to create primitive local managers from scratch. Since, as mentioned above, the distinction between the network data manager and the local data manager is not sharp, primitive local managers would be reasonable if the network manager is more sophisticated. The major disadvantage to this approach is that more original software would have to be created than if existing data managers are adapted. The advantage is that the local managers could be designed and built specifically for the task at hand. This means, for instance, that the local manager could be better integrated into the network manager.

There is considerable debate as to which alternative is more valuable to the research program. An effort is scheduled to resolve this question in the two month period following submission of this research plan document and prior to beginning the research program.

Intelligent Terminal

The purpose of the intelligent terminal effort is to take the next evolutionary steps in distributed data management. By the early 1980's it will be technically and economically feasible to provide each user on a network with powerful terminal resident processing capabilities dedicated to him alone. The intelligent terminal option will provide a mechanism for exploring the impact of that technology. Preliminary research indicates significant improvements in distributed data system performance, cost, and survivability when intelligent terminals are incorporated into the architecture of such systems. Other benefits accrue in the human engineering area. The power of the intelligent terminal may be used to mollify the cumbersome idiosyncracies of network and host control languages. The only way to assess the significance and impact of the intelligent terminal concept is to build one and experiment with it.

The present concept is to utilize a minicomputer system like the Digital Equipment Corp. PDP-11 or the Data General Corp. Eclipse with 64K (16 bit) words of main memory, a floppy disk secondary storage device, three high-speed plasma panel display devices with touch input, a keyboard and network communication interfaces. The software would consist of an operating system and various utilities (e.g., display service routines, communications interfaces, and basic device support routines), so that applications experiments may be simply implemented.

The system just described is a large and expensive one. Its cost would be in the $80,000 range today and is estimated at $10,000 to $15,000 in five years. The purpose of considering the large system at this time is to permit experimentation with a realistic vehicle that might be introduced into the production WWMCCS community early in the 1980's. Smaller, less capable, experimental systems can be configured in the $20,000-$40,000 size.

The following discussion describes some of the capabilities and features that can be addressed in an intelligent terminal program.

Distributed data management system cost and performance improvements. Some preliminary work has been done at Harvard with the use of intelligent terminals to access a large host. The interesting result is that, even at today's prices and performance figures, using relatively simple data sharing and processing strategies, the Harvard work reported a 20-30% improvement in response time and 20-30% reduction in overall system costs. The principal reason seems to be that mini- and micro- computers can perform most I/O functions as fast as large scale computers while tying up a much cheaper processor. A simple strategy for distributed processing in an intelligent-terminal-based DMS is to give the terminal some small amount of local storage (e.g., a floppy disk) which is a local buffer to the much larger data base at the remote host. The first time a set of data is accessed, it must be fetched from the remote host. If at a later time, however, that set must be examined again, it can be retrieved from the local buffer at a small fraction of the cost of retrieving it from the remote host. The response comes back much faster because the terminal resident processor is a dedicated resource and is immediately available to respond. Costs are reduced in terms of processor consumption at the remote host and communication line traffic. The one thing that a large remote host typically does not do well is start I/O operations and receive the interrupts when they are completed. Since the intelligent terminal can reduce that kind of load on the remote host, the remote host finds itself more available to perform those computing functions which it does do well and cost effectively. In the WWMCCS environment this reduces the load on the main hosts and improves their responsiveness. This can help to avoid the cost of upgrade as host load increases.

Failure detection and recovery aids. The intelligent terminal is also a logical place to provide failure detection and recovery aids. Checks can be put in to monitor the phone line, check front-end operation, host operation, etc., and automatically dial around malfunctioning components. With the addition of local storage, plus appropriate data management protocols, the intelligent terminal can contain enough status information to facilitate system restart at a new host (after primary host failure) without user intervention.

Improvement of communication line efficiency. Since there is processing power local to the terminal, it is feasible to provide automatic data compression facilities to improve the speed of transmitting data between host and terminal and to let the terminal do the final formatting of reports for display. The compression capability is especially valuable for reducing the bandwidth requirements of imagery data (e.g. aerial photos).

Security. The processor can be used to provide cryptographic facilities at the terminal for both end-to-end and step-by-step encryption. Also, sophisticated authentication facilities can be supported.

Survivability. If we imagine an intelligent terminal used as a command terminal for each major organizational unit in a military environment and if we provide enough local storage at the terminal to maintain the data relevant to that unit, then a very interesting situation develops. The terminal communicates with a large and sophisticated data system resident at a remote host where the data for all units are gathered. If, however, host or communication failure occurs, the local terminal has most of the data about the local unit and a minimal system to access that data, and permits continued, but degraded, operations. After a host failure, a new host can be designated to perform large system functions. The command terminal can automatically forward its data base to that host and, in effect, provide its own backup for the remote host.

If there is command terminal failure, the data base there can be reloaded from the remote host. As can be seen, a technology that is desirable because of its cost and performance parameters also provides a very natural backup mechanism (as contrasted with the more artificial policy of deliberate over-buy of resources to back up malfunctioning resources). This technological approach is an inherently survivable approach.

Touch panels and graphic display. Touch-panel/graphic-display capabilities will permit the testing of touch/display command languages which include very rapid response menu selection and command input based on touch sensitive flowcharts. In addition, the graphic facilities should permit the display of bar charts, pie charts, and line graphs in addition to standard tabular reports. Animation capabilities to blink an entry or to blink a box around a column or row in a table may also be useful for displaying high priority items.

Multiple displays. Three display screens can be provided. One directly in front of the user, one to his left and one to his right. This will facilitate experiments with automated user assistance (help facilities), real-time monitors and scratch-pad facilities. For example, the user's line of sight and principal concentration is on his front panel. For whatever part of the system he is currently using on the front panel, the appropriate help text for that part of the system may be scrolling by on the right-hand panel. Thus, if the user has a question he need only look to his right to find the "user manual" always open to the right page. At the same time, the left-hand screen could be used to store reference information (e.g., previously prepared reports and graphs or a real-time situation display) to which he has occasional need to refer, but which would only clutter up his primary screen or, worse yet, his train of thought (e.g., if a split-screen technique were used on the front screen and that information was always in sight).

Transparency aids. Experiments can be conducted in system transparency and language translation. For example, with CPU power in the terminal, it should be possible to develop a touch/display application-oriented command language which the user might prefer. The terminal could then translate that language to a more complicated WWDMS language syntax for processing by a large GCOS host. The imbedded processor could also be used to hide system-dependent idiosyncracies. For instance, each system on a heterogeneous network (e.g., the ARPA network) would be expected to have a different technique for transmitting messages between terminals, sending mail, finding out who is logged into the system, and accessing the system help files. The processor imbedded in the terminal, with adequate programming (probably loaded over the network, dynamically, from the host that it is accessing), could provide a consistent interface to each of these capabilities on a wide variety of hosts. If automatic dial-out capabilities were provided in the terminal, the user would be able to log into his terminal and have the terminal automatically direct dial a host or a network front-end, log into the network, Telnet to an acceptable host running the sub-system the user wishes to access, log into that host, and automatically initiate the sub-system. Thus, for example, the user of a JOPS capability might only log into his terminal and type the name of his application package. The terminal would take over from that point in locating the application package and connecting him to it.

Technology Transfer

The goal of the technology transfer task is to aid in the dissemination of recent research results to the WWMCCS community. In addition, by maintaining familiarity with the realities of the PWIN environment, we will be better able to make the rest of the research program relevant to the WWMCCS community. Technology transfer is too often underemphasized in research programs, to their detriment. We plan four major technology transfer vehicles.

1. There will be event-driven reports. When a particular portion of research has reached a suitable point, a report describing it will be prepared. These will supplement the periodic progress reports.
2. A limited consultation service will be provided directly to WWMCCS end-users, with JTSA management of the effort.

3. Technical assistance will be provided to JTSA on networking problems as required and mutually agreed.
4. It is hoped that the periodic contract reviews can be enlarged to include a more tutorial discussion of recent research results.

## Research Plan

### Overview

In this section, more detailed approaches to the individual research areas of figure 1 are discussed within the broad programs presented in the research strategy section. To aid the reader, a common format is adopted:

Purpose and scope. In broad terms what the topic is about, including definitions if necessary.

State of the art. A discussion of what has been done on the topic, and what research is known to be progressing elsewhere.

Applications orientation. What JTSA will be able to do with the research products; alternately, what applications will be difficult or impossible if the research is not completed.

Proposed research approach. The methodology and strategy for the planned research.

Anticipated results. Basically, what can be expected from the research, and approximately when results should appear.

This research plan covers a relatively long period in a field where startling developments occur quite frequently. Thus the reader must keep in mind that the plan is simply our best estimate on the problems. The plan identifies the major topical areas and the most essential dependencies between them. It is intended to serve as a chart would serve an experienced river pilot, not as something to be blindly obeyed but as a framework within which to apply reasoned, professional judgements.

### Synchronization

Purpose and scope. When two or more processes have to share the same resource, some form of synchronization is required among them to effect the sharing. In a network environment these processes may reside on different hosts. The majority of classical synchronization methods were designed for single-site computer systems and require a shared memory interface to be effective. Either the classical synchronization techniques must be upgraded for network implementation, or new techniques must be developed to permit the orderly sharing of network resources.

Distributed data base system synchronization presents additional challenges. A data base system has many more resources requiring synchronization than an operating system. The synchronization must occur at the logical (rather than physical) level because there may not be a uniquely identifiable physical resource corresponding to a given logical resource. Gross synchronization (e.g. file lockout) will not be well suited to a network environment because the time required for network process synchronization will mean that the file will be locked for long periods of time.

Applications orientation. It is practically impossible to consider a general distributed data management system which does not use synchronization. If updates to a distributed data base are not synchronized, the data base may become internally inconsistent. For example, if one process sets a field to 100 and another process sets the same field to 150, then those updates must be synchronized so that they are performed in the same order on all copies of the distributed data base. Otherwise, some copies of the field would have the value 100 while other copies of the supposedly identical field would have the value 150. Synchronization is required for any other application which is implemented by having cooperating processes running on different hosts (e.g., workload sharing). Thus, synchronization is essential to nearly all serious applications of the PWIN/WIN.

State of the art. The majority of the work on synchronization relates to the problems of synchronizing the processes running on a single computer. In this case, inter-process communication is accomplished via shared memory. The most common techniques use some form of "read-modify-write" instruction (e.g., test and set) to implement system synchronization primitives (e.g., lock & unlock, block & wakeup, P&V). Very little has appeared on practical techniques for efficient cross-network process synchronization. In a single computer situation, the decision by a process of whether to proceed or to wait at a synchronization point can be made in a few microseconds. In a network environment the same decision may take a second, because several messages must be transmitted and received via the network in an attempt to simulate shared memory. This enormous real-time difference for interprocess communication makes classical synchronization ineffective and unusable on a network. Even if synchronization could be implemented with a single exchange of messages, the delays are still very long relative to single sites (tenths of a second as compared to microseconds). Techniques must be developed to minimize the frequency of synchronization far below traditionally acceptable levels. It is also not at all easy to produce correctly synchronized processes. Subtle differences in implementation or in the use of synchronization primitives may yield disastrous results (e.g., deadlock, lack of synchronization).

Proposed research approach. The predominant cost in network synchronization is the delay experienced by messages. Thus the first approach taken will be to choose those classical synchronization techniques which require the least communication overhead and adapt them to the network environment. This generates basic problems of naming the cooperating processes, the shared resources and the memory cells used to control the synchronization. In addition, problems of security and resiliency are raised for the production network user. For example, on a network, one host might acquire a resource at a second host and synchronize that acquisition with all other hosts on the network. If the first host then crashes, the state of the network is uncertain. If it is not possible to release the resource owned by the crashed host, then all hosts waiting for that resource are deadlocked. This problem doesn't even exist in a single site system since nothing can proceed if the site fails.

The second approach taken will be to carefully study the synchronization problem, especially from the data management viewpoint, to minimize the requirement for synchronization. For example, if the DMS were designed so that the only updates permitted on fields were increments and decrements, then updates could be performed in any order and the field value would be the same after all updates were processed. Of course, this is a very specific example of dealing with numeric fields. However, a study of data types and the operations on those data types which do not require synchronization is in order. The synchronization-free data organizations identified in this way will then be examined to determine which data bases they can be applied to. Some possibility exists that hybrid approaches may be very useful. In a hybrid approach, synchronization-free operations would be used wherever possible. When absolutely required, an operator which needs synchronization would be used and the necessary overhead incurred.

Anticipated results. Since the synchronization problem is so basic to the fundamental questions of WIN use, usable results must be generated almost immediately. We would anticipate that an elementary synchronization mechanism could be created by the end of the second quarter of the project. Naturally this elementary version would undergo refinement as the project progresses.

## Name Space Management

Purpose and scope. All of the resources, files, etc., in a network must be assigned unique names so that they may be referenced unambiguously. Although there are no technical barriers to solving the problem, some thought must be given to the design of a satisfactory naming scheme. Some of the questions which must be addressed are:

1. Should names all be assigned by one central name assigner?
2. If not, what rules should the individual assigners follow to avoid duplication?
3. If a named entity is destroyed, can its name be released for future use without introducing possible confusion?
4. How are the various naming structures used at different sites in a heterogeneous network to be reconciled?
5. Should names implicitly or explicitly describe the location of the named object?

Applications orientation. Even a single-site computer system must have an arrangement for ensuring that no two entities have the same name. A naming scheme of some sort is therefore an essential ingredient of a viable network. A study of various alternatives is immediately applicable to the PWIN, in that some alternative must be chosen, and preferably on a rational basis.

State of the art. As noted above, no technical barriers to devising an effective naming scheme appear to exist. Some thought has gone into the problem of making names unique throughout a network. One of the alternative schemes already suggested may well turn out to be all that is needed. No scheme, however, has undergone the rigorous network testing necessary to demonstrate its effectiveness.

Proposed research approach. We propose to test alternative naming schemes in the context of the experimental system. Using that system as a test vehicle for promising, alternative schemes is a natural approach which (with little risk) should lead quickly to readily applicable guidelines for naming resources in the PWIN.

It should be noted that the naming problem is one which may have to be readdressed later in the project. As our understanding of the complexity of a network data system increases, we may find that a scheme which worked well in an unsophisticated environment is no longer adequate. Again the chief tool in identifying such a problem and attempting to solve it is the experimental system, which will grow in completeness and complexity as the research progresses.

Anticipated results. This problem can be addressed in the first year of the project and will probably yield usable results, in the form of recommendations for the choice of workable, convenient naming schemes, by the end of the first year of the project.

## Network File Allocation

Purpose and scope. For a single-processor system, the problem is to determine how best to allocate the files needed by a program (or programs, in a multiprogramming environment) among the various memory devices available. In a network, the problem broadens to include the possibility of distributing files (including backups) among the various network sites. Guidelines must be developed for good network allocation strategies. The allocation strategies will form an important part of both manual and self-tuning distributed data management systems.

Applications orientation. As mentioned above, file allocation is an integral part of a distributed data base system. File allocation policies represent the grossest tool for system tuning. The choice of file location(s) can have dramatic impact on nearly every facet of network performance. Thus, file allocation is of immediate relevance for any proposed network applications.

State of the art. Some work has appeared on the file allocation problem. Its obvious similarity to the classical warehouse location problem of operations research has probably helped to motivate much of the work. Many of the published file allocation papers ignore critical factors (e.g., synchronization) which seriously degrade their usefulness. In a very recent paper [K.P. Eswaran, "Placement of Records in a File and File Allocation in a Computer Network," Proc. IFIP 74, North-Holland Publishing Co., pp. 304-307] it was shown that the optimal file allocation problem (i.e. the problem of finding the "best" allocation) is polynomial complete. This formal result says that if we could solve the optimal file allocation problem in polynomial time (i.e. in a length of time which is at worst a polynomial function of the "complexity" of the problem) then a number of notoriously difficult problems could also be solved in polynomial time. Its practical impact is that for reasonable sized file allocation problems, no simple enumeration scheme will give the optimal answer. Rather, we must rely upon heuristic techniques and settle for "good" file allocation strategies. The problem is compounded since the input data (measurements) may not be good enough to warrant the expenditure necessary to find a "best" allocation.

Proposed research approach. Since the basic idea of file allocation is well understood, we plan initially to adapt existing work where possible. As the research progresses, and our knowledge of the realities of distributed data management increases, we may find that much of the previous work done by others is no longer applicable to the problem. It appears, for example, that available CPU capacity is a significant consideration in file allocation, but few authors have used CPU capacity in their allocation optimization strategies. The relationship between file allocation and load balancing should also be studied, since the two topics are facets of the same problem.

Anticipated results. Some results in the file allocation area are anticipated early in the project. These initial results will likely be extensions of existing work. They will, however, give some guidance to the designers of distributed data systems. As the project continues, these results will undergo refinement and analysis to improve their usefulness.

Multiple Copy Management and Backup

Purpose and scope. One advantage of a distributed data base is that copies of segments may be created at different hosts to reduce bottlenecks and to improve response time and reliability. It is a non-trivial task to operate a multiple copy data base. Activity between copies must be coordinated so that they remain consistent. Copy access strategies must be developed to facilitate failure detection and to provide for a smooth transition to backup copies with little or no human intervention. The purpose of this research area is to develop the basic naming, structuring, access, and backup strategies required to support a distributed data base.

A sufficient (but not necessary) condition for consistency between copies is that the updates be applied to each copy in the same sequence. Other problems occur when an attempt to balance the load on each copy is made by farming out the query load. If this is done randomly, it may not be possible to process the query load as efficiently on the distributed data base as it would be to process it on a non-distributed data base. This could occur when, for example, a number of queries reference a single block of physical storage. In a sophisticated non-distributed data base, it is likely that only one I/O operation would need to be performed. In a distributed situation with random query distribution, each host might have to duplicate the fetching of that block. Thus, considerable attention needs to be devoted to intelligent query management to maximize the system's performance.

Large data networks like the WIN require significantly more sophisticated backup techniques than are normally employed in computer systems. Given the large number of data bases involved, recovery from a single host failure could be exorbitantly time consuming if human intervention were required to switch to backup copies on other hosts. In a 35-host WIN, even if each host is crashing only once per day, on the average every 40 minutes some host will be crashing and must be backed up. Furthermore, in a network with 35 hosts, multi-host crashes, even in the middle of backup procedures, can be expected to occasionally occur. Conventional single-host backup techniques are operator oriented and must be improved on and automated for the network environment. High reliability applications are usually supported by duplex systems with one system standing by as hot backup or with both systems simultaneously performing the task and checking each other's operations. The network is inherently an n-plexed environment with individual hosts capable of using their excess capacity in a natural standby mode.

Application orientation. This research is directed at analyzing the most basic aspects of distributed data management. The results can be immediately applied by JTSA in the analysis of functional specifications and proposals for PWIN/WIN distributed data management.

State of the art. Very little formal work has appeared in the literature about multiple copy management. Many authors seem blissfully unaware of the update synchronization problem, and how vexatious it can be to solve in a network. It is also not at all clear from any published work whether careful analysis of the costs and benefits of distributed data management has been done. Many papers have an implicit, a priori assumption that distribution of a data base is desirable and then proceed to analyze how best to do it. Our preliminary research studies indicate that synchronization and backup are major problems and the implicit assumption that distributed data bases are desirable should be questioned for each application.

Proposed research approach. The research will be approached in three phases: synthesis, analysis, and test.

1. In the synthesis phase, the design alternatives of multiple copy management will be informally formulated. This was partially accomplished in the Scenario Report [CAC Doc. No. 159, JTSA Doc. No. 5506], and continuing effort is warranted.
2. In the analysis phase, the design alternatives will be more rigorously characterized and then evaluated in the framework of the analytic model.
3. In the test phase, detailed algorithms will be developed and implemented for those alternatives which appear most promising. This work will be part of the experimental system activity, and these algorithms will form the heart of that system.

Continuing feedback and evaluation is anticipated between the analysis and test phases.

Anticipated results. The major results will be reports analyzing the various design alternatives which have been proposed for multiple copy management. The analysis will include such factors as relative cost estimates, reliability and availability estimates, and anticipated loads on the hosts and the network. Demonstrations will be available for alternatives implemented in the experimental system. For the most viable alternatives, we plan to suggest methods for choosing the optimal ways in which to use them, such as where copies should be located, master-slave relationships, etc., under externally imposed boundary constraints (e.g., command prerogatives). Thus, the realities of the PWIN/WIN environment may be incorporated explicitly. The combination of the analytic modeling and the experimental system should yield results of high confidence and significant utility to JTSA in planning future WIN applications.

## Deadlock

Purpose and scope. A deadlock (some authors have used the term "deadly embrace") is a situation in which the progress of two or more processes is mutually and simultaneously prevented. Deadlocks result from uncontrolled competition for a set of resources. The opportunity for a deadlock to arise increases rapidly as the size of the set of resources, the number of requesting processes, and the frequency of resource allocation increase. A data base management system is often a highly deadlock-prone environment, since the set of resources is composed of (possibly small) fragments of a large data base, the number of users may be high, and the allocation/deallocation of resources occurs very frequently. Conventional solutions to this problem involve rapid communication (e.g., shared memory) between the processes using the data base. Due to large (order of tenths of seconds) communication delays, a distributed data management system may not be able to use conventional deadlock prevention/recovery techniques without degrading service intolerably.

Applications orientation. Deadlock is a problem in any data base management system which provides an access type which is exclusive (non-sharable) and cannot be pre-empted. Every multi-user data management system must have such an access type to prevent destructive interference between two or more users who are trying to modify the data base at the same time. To prevent deadlock, some systems impose restrictions on the use of exclusive access. These restrictions may make some data base operations difficult or impossible. For example, suppose it is necessary to change the value of a field in two records belonging to different files of a multi-file data base. This either requires exclusive access to both files at the same time or else there is a time (after one record has been changed and before the other one has been changed) that the data base is inconsistent. In general, the location of the second record might not be known until after the first record has been accessed. A system that permits only one file at a time to be "locked" makes it impossible to perform this update. Thus, the study of the deadlock problem has immediate application to real-world WWMCCS data base problems.

State of the art. The necessary and sufficient conditions for deadlock are well understood. However, the implications of these conditions in data management systems are much less understood. There are well known methods for deadlock prevention in single site systems. Unfortunately, these methods either tend to be inefficient in their utilization of resources or they impose unacceptable constraints on the application environment (e.g., identical ordering of resource allocation and deallocation for all processes). The deadlock problem in the network environment is more severe than in the single-site environment due to interprocess communication delays and the larger resource set. Unfortunately very little literature has appeared on the problem of deadlock in networks and distributed systems.

Proposed research approach. The deadlock problem will be addressed in three phases.

1. The existing literature will be analyzed and, where appropriate, the current understanding of deadlock will be reformulated to include network considerations.
2. Network-relevant detection and recovery strategies will be formulated in parallel with the strongly related synchronization and backup and recovery strategies. These strategies will then be refined within the modeling activity.
3. The more favorable approaches will be tested by implementation on the experimental system. Based on implementation success and failure, reformulation and further modeling and testing may be carried out.

Anticipated results. The major anticipated results are cost-effective techniques for deadlock prevention or deadlock detection and recovery in distributed data management systems. Deadlock is a very basic system consideration. At the lowest level the results of deadlock research will be seen as refinements to backup and synchronization algorithms. Results will also be strongly reflected in processor allocation (for load balancing) and network file allocation techniques.

## Data Clustering

Purpose and scope. Data clustering is the technique of grouping items for improved retrieval efficiency. The clustering algorithms themselves must be efficient, applicable to very large data collections, easily implemented, and based on sound theoretical principles (i.e., demonstrations that they do accomplish the desired improvement in efficiency). We propose to develop a set of clustering algorithms:

1. for identifying the need for clustering (or re-clustering),
2. for determining what clusters should be formed, and
3. for carrying out the clustering.

Such a set of algorithms would form an important component of a self-tuning data management system.

Applications orientation. Even partial results (i.e. a far-from-optimal clustering scheme) can provide dramatic improvements in data management system performance. In general few of the fields in large records and few of the records in physical blocks are actually used to answer a query. Yet an entire block must be transferred whenever one of the fields in one of the records of that block is accessed. Commonly, only 1% to 5% of the data so transferred is actually used. As a result, ADP systems are extremely I/O bound and a large number of I/O operations must be initiated to access all of the blocks required to respond to one query. In the typical ADP shop, approximately half of the CPU is devoted to initiating and terminating I/O operations. Thus it is clear that useless transfer overloads the CPU as well as the I/O equipment. Blocking (or clustering) the data for higher utilization per transfer can provide substantial reductions in costs, CPU load (since the number of I/O operations is reduced) and response time (since I/O transfer is reduced).

State of the art. Most work on data clustering has been done in the context of document retrieval. Documents with similar content are clustered together and retrieved as a block. This idea has been extended to more general collections of data, in which the clusters are based on the similarity of attributes. In this case, one does not retrieve a whole cluster, but restricts the search for relevant items to a particularly promising cluster or group of clusters.

A recent development has been the clustering of data on the basis of query patterns - i.e., the grouping together of items which show a high probability of being retrieved together. Under the present contract, we have contributed to this technology by developing an algorithm for what we call dynamic query clustering [Preliminary Research Study Report, CAC Doc. No. 162, JTSA Doc. No. 5509]. Although details remain to be worked out, this algorithm promises to provide an efficient means of collecting the statistical data on which clustering according to query patterns should be based.

Proposed research approach. As noted above, we have already obtained some preliminary research results. The Preliminary Research Study Report contains a fairly detailed discussion of what needs to be done next. The first step is a thorough study of various options and parameter choices in the query clustering algorithm. This study will make heavy use of the experimental system to be designed early in the project. Some modeling may also be feasible and desirable, although this point is as yet unclear. Once the query clustering algorithm is defined with more precision, we must proceed to a two-pronged effort:

1. to determine how query clusters may be used in an algorithm for deciding when to cluster, and
2. to examine various alternatives for clustering the data on the basis of query patterns and to design algorithms for carrying out this task.

The largest element of risk involved in the project is that it may be difficult - perhaps even impossible - to design algorithms which are not too costly and inefficient when applied to very large data bases. Cost and efficiency must be an overriding concern throughout our study if the results are to be optimally useful to JTSA.

Anticipated results. The research effort has already begun. The query clustering scheme can probably be refined into a usable, efficient algorithm by the end of the first year of the project. This will, however, depend upon how quickly the experimental system is brought up to a level capable of handling the necessary experimentation.

Integration of the query clustering algorithm into a complete package of algorithms for deciding when and how to cluster is a goal which we hope to attain within the three-year effort. But, as we remarked above, the project is not risk-free. On the other hand, we envision no serious difficulties in getting "good" algorithms which can be applied, at least in a limited context, to get limited (but not negligible) improvements in cost and response time for heavily used portions of a real data base.

## Automatic Data Structure Design

Purpose and scope. The problem addressed here is that of designing a mechanism to automatically select an optimal (or near optimal) structure (or organization) for the data. This includes not only the choice of physical structure (e.g. whether the data should be stored in tabular or hierarchical form), but also the indexing or hashing schemes to be used in accessing the data and the clustering and file allocation to be carried out. As a first step in solving this problem, guidelines must be developed for making the necessary choices. The guidelines must then be automated. Finally, the restructuring problem, or the automatic translation from one structure to another, must be solved, in order for the automatic selection mechanism to be of real use. (This last problem, however, we consider to be part of the development of a self-tuning distributed data base management system. See below.)

Applications orientation. Retrieval efficiency is strongly tied to data organization. In a completely unorganized collection of data (e.g. a linear list with no indexing), the whole collection must be searched to answer a query. Any sort of indexing speeds up retrieval; but then questions arise as to which attributes should be indexed, and how the index itself should be structured. If the power of the computer can be used to help make analyses (e.g. of usage patterns) and to help decide which structures will yield efficient retrieval, impressive improvements in response time will follow.

State of the art. A considerable amount of work has gone into the development of guidelines for optimal or near-optimal choices for some aspects of data structuring. As an example, we cite the recent work of Rothnie and Lozano [Communs. ACM 17 (1974), pp. 63-69] on constructing an optimal combination of file inversion (indexing) and multiple-key hashing. File allocation has been studied, but as was discussed above, much needs to be done to develop efficient heuristic methods for file allocation. The automatic clustering problem (also discussed in more detail above) is one which we have begun to study and hope to obtain usable results on during the first year of the proposed project.

Attempts to implement automatic structuring have been carried out, but only on a small scale. That is, only a limited number of options have been considered by the system, and the system has only been implemented for very small data bases. A typical approach has been to notice (automatically) that a particular attribute is referred to extensively and to build an extra index (inverted file) based on this attribute. Clearly such an approach addresses only a small portion of the overall automatic structuring problem. These limited studies do indicate, however, that a more ambitious effort is feasible and may yield a large payoff.

Proposed research approach. A first step is the development of guidelines for choosing among various structures and indexing methods. Much information already exists in the literature on this subject. We intend to examine various known approaches and test them using the experimental system and possibly the model. Developing a feel for what structure is optimal (or near optimal) in all possible situations is probably an unrealistic goal. By designing the model and the experimental system with JTSA's needs in mind, however, we should be able to produce guidelines applicable in the WWMCCS environment.

Our research effort in file allocation and automatic clustering should lead to heuristic algorithms for decision-making in those areas. Results in those areas will then be applied to the automatic structuring project. Integration of all results obtained and of all algorithms developed into an automatic structuring system will be the final phase of the project.

Anticipated results. Fairly early in the project we should develop some WWMCCS-relevant guidelines for structure choices. These should be immediately useful to the WWMCCS community even though they must be applied manually. During the second contract year, we expect to develop a reasonably good understanding of the various aspects of automatic structuring. This work will include the construction of decision algorithms which again may be immediately applied to provide for limited automation. The whole effort will be a continuing, high-risk project, however, and it is likely to take many man-years of effort before the solutions to the various partial problems will be integrated into a successful, coherent, automatic structuring mechanism.

Distributed Data Management System - Manual and Self-Tuning

Purpose and scope. We are here dealing with the ultimate goals of the research effort. The first goal is to develop a better understanding of various aspects (discussed above) of distributed data management. As results of the research are worked out and tested, they may immediately be incorporated into working distributed data base management systems. For a test vehicle a "manual" DDMS will be developed. The manual system will test the latest DMS technology as it is produced, but may require human intervention at critical points. For example, while backup procedures may be automatic, a human might be required to choose and commit major resources for the backup task before the procedures are invoked. Alternatively, clustering algorithms at first might only produce summary statistics which identify data clusters. A human would then be required to determine the cost tradeoffs of restructuring the data to conform to the identified clusters.

A later goal is the development of an automatic, or self-tuning, distributed data management system. It is anticipated that the manual DDMS will evolve into the self-tuning system. Ideally, this would be a system which automatically responds to user needs, restructuring the data, building indexes, clustering the data and distributing the clusters automatically, etc. Such a system is an ideal which will not be readily attainable. In a three-year, limited-manpower effort, the most that can be expected is the development of some components - such as automatic clustering algorithms - of such a system.

Applications orientation. The goals discussed above may be restated as the development of techniques for the dramatic improvement of the response time and ease of use of the WWMCCS data bases. The automatic, self-tuning features could be particularly important in a real-world, crisis situation when responsiveness of the data management system to immediate (and rapidly changing) user needs and system bottlenecks could literally mean the difference between life and death.

State of the art. Very little work has been done on the hard problems of distributed data management. Most papers written on that topic have been of the "concept" type - pointing out problems but doing little to solve them. On the other hand, the area of data management itself is fairly well developed. Indeed, the extensiveness of the literature and the large number of alternative techniques to choose among have themselves become problems for the data manager. Some areas are, of course, better understood than others. We have already discussed (under separate headings, above) the state of the art of some of the key areas which go to make up a complete, distributed data management system.

Proposed research approach. Nearly all of the research efforts described in previous sections play a role in our work towards the goal of developing distributed data management system technology. The network-oriented problems of synchronization, deadlock, multi-copy management and load balancing must be solved before distributed data management can become a reality. In addition, all of the research into data structuring, file allocation, clustering, etc., is helpful to the design of a good manual data system and is essential for an automatic, self-tuning system.

A key part of the self-tuning system will be an automatic structuring mechanism. The research required before such a mechanism can be possible has already been discussed. We have also noted that the research is high risk and may not be successful in the near term. Another aspect to automatic structuring is the translation problem - i.e., the transformation of data in one format or structure into another. This problem has been under study by Merten, Fry, et al. at the University of Michigan. We assume that further study will form a natural part of their future work on the data restructuring problem and will provide input to this program.

Other problems that we do not plan to directly address - e.g., the study of network protocols - also arise in distributed data management. It is therefore unclear at this stage how soon all the necessary technology for automatic distributed data management will be available. Since the technology is, in many areas, in a very primitive state, it is impossible to predict how much effort - or research along what lines - will be required to integrate the various technical areas into a coherent whole and to fill in the gaps which are almost certain to appear.

Anticipated results. Much of the technology needed for a good, manual, distributed data management system can probably be developed during the course of the three-year effort. As was pointed out in previous sections, solutions (or at least partial solutions) to some of the outstanding networking and data management problems should be forthcoming within the first year or two of the contract. Each technological improvement we obtain may be applied as soon as is feasible to the WWMCCS systems. As technology transfer takes place, the WWMCCS community will therefore see not a suddenly-appearing, perfect, distributed system, but a continual and gradual improvement in their facilities.

This improvement should include the eventual introduction of automatic features - steps along the way to the ultimate, ideal, self-tuning system. But, as indicated above, we are unable to predict when the extensive technology development required for such a system will be completed.

## DDMS Workload Sharing and Load Balancing

Purpose and scope. The problem of load balancing or workload sharing is to efficiently utilize the resources in a computer network. In network environments it is common to see some hosts working at maximum capacity and still not being able to keep up with their load while other hosts have substantial excess capacity. Further, this is a highly dynamic situation with the load on each host varying rapidly. It is the goal of workload sharing to prevent processing bottlenecks by shifting tasks from an overloaded host to a less loaded host and to improve the response for high priority tasks by also balancing the priority processing load. It is the goal of this research to create algorithms for semi-automated workload sharing, to study these algorithms and to suggest load balancing strategies which should tend to maximize overall network throughput. As a first step, some load balancing capabilities can be implemented within the manual, experimental DDMS to automatically shift processing loads among the network hosts on a query-by-query basis. As a later activity, the development of more general workload sharing capabilities can be undertaken for applications external to the DMS.

Applications orientation. The current PWIN workload sharing mechanism is a cumbersome remote-job-entry system in which a large amount of human intervention is required. Some WWMCCS hosts are already substantially overloaded while others are relatively lightly loaded. With the advent of distributed data bases, additional dimensions to the problem are created. As is discussed in the section on multiple copy management, an unsophisticated workload-sharing/load-balancing scheme can actually hurt system performance by requiring duplication of the same effort at several hosts. Thus, meaningful workload sharing and load balancing techniques are of both immediate and longer term relevance to the PWIN/WIN community.

State of the art. Technology in workload sharing is almost non-existent. There are severe practical difficulties to implementing even a crude system. Some classes of problems (such as incompatibilities of programs and data) which are severe in a hetrogeneous network like the ARPANET are reduced in a homogeneous network like the PWIN. Although some well-thought-out proposals have been made for ARPANET workload sharing, none has achieved acceptance by the community. Load balancing depends on a practical workload sharing system, a workable definition of load on a system and a means for measurement of the load. At present none of these elements exists in a usable state.

Proposed research approach. The research approach may be divided into three sections.

1. There is the general question of what is load and how can it be measured. Both theoretical models and a supporting program to produce empirical measures of potential load measurement

components for a wide variety of machines are required.

2. A theoretical investigation of the mechanics of workload sharing and load balancing is required. We plan to consider when and how a given portion of the workload on one host should be moved to another host.

3. The work must be applied to specific problems of distributed data management.

These specific problems can be broadly separated into two groups: static questions of structure and dynamic questions of management. The static problems are, in general, related to file allocation, which was discussed previously. The dynamic questions relate to where a given query should be processed. The workload to be shared is the set of queries against the distributed data base. (Each copy of a file must process all updates, so no sharing of the update load is possible.) In the section on multiple copy management, it was shown how a naive query distribution scheme might actually reduce overall network throughput because I/O operations would be unnecessarily duplicated on several hosts. The query load is highly dynamic and unpredictable. Thus, it is unlikely that the system would have the time to search for optimal distribution of the queries. Rather, our research will be directed at finding effective, low-overhead query management schemes.

Anticipated results. The research in workload sharing is planned as a continuing effort throughout the life of the project. A preliminary model of load and a basic set of empirical load measurements developed in the second year of the program should produce a working definition of load and some initial results in the area of workload sharing, such as the basic algorithms and mechanisms. The ability of a system to balance its entire load automatically is a longer-range technology. Load balancing as applied specifically to query management will be an intermediate term result, although the theory will likely precede the demonstration by two to three quarters. If successful, this special form of load sharing within the DMS could be transferred to the WIN with high impact since data management is anticipated to be the major WIN load.

Optional Intelligent Terminal Program

Purpose and scope. Four research areas are considered in the intelligent terminal program:

1. intelligent terminal experimental vehicle,
2. terminal resident DMS components,
3. man/machine interface criteria, and
4. prototype intelligent terminal.

The purpose of research in these areas is to develop intelligent terminal technology in support of WWMCCS distributed data management systems and also to improve the human interface to WWMCCS application systems. Basic concepts of intelligent terminals as they relate to data management and the human interface have been discussed previously in the section on research strategy.

Applications orientation. At present many of the existing hosts on the PWIN and the proposed hosts on the WIN are saturated and provide poor response. The cost to the Government to upgrade these facilities is substantial. In a data management environment, like the WIN, terminals with embedded processors and modest local storage can significantly offload the larger hosts. Those functions which the large host performs inefficiently and at high costs can be performed for less cost at the terminal. Thus large host upgrades can be avoided or reduced. At the same time, the dedicated local processor in the terminal can provide responsiveness to the terminal user which is economically infeasible with conventional terminals and large remote hosts.

During the survey of PWIN systems and interviews with WWMCCS users, it was clear that the human interfaces to existing and proposed WWMCCS network systems were programmer oriented and required extensive training. Furthermore, in order to maintain familiarity and facility with each of the WWMCCS systems, constant use is required. In normal peacetime operations only one shift of operators has this day-to-day usage and maintains an acceptable skill level. Thus the system becomes unusable in crisis situations when a command center is manned on a three shift basis, but only one shift has the required skill level. Experience has shown that in those situations telephone, facsimile and other easy-to-use facilities are employed to avoid fumbling with the computer system. The intelligent terminal concept has significant capability to improve the human interface. At the simplest level the terminal can implement a better human interface, using technologies like application oriented touch/display command languages. The terminal can translate these user oriented command languages into the programmer oriented job control languages and application package interfaces. At more sophisticated levels the terminal, with its embedded processor, can cooperate with a large host application package designed to take advantage of the terminal capabilities and produce a level of accommodation to the individual user that is not feasible with present technology.

As a side benefit, an intelligent terminal with embedded data management abilities integrated with a large host data management system provides a very natural backup capability and can support survivable operations. Although survivability is not a major WWMCCS network goal at present, significant concern was expressed by the user community about the survivability question and the growing dependence upon computer-based command and control systems.

State of the art. As was indicated earlier in the section on research strategy, preliminary work has been undertaken at Harvard on the use of multiple minicomputers to perform some minor data management front-end activities in support of a large host data management system. A one-on-one, CPU-to-user approach has been demonstrated at the CAC during the preliminary research studies. Outside of these two activities, virtually no work has been done in the use of intelligent terminals applied to data management. The human interface area is more active. ARPA has the large National Software Works program aimed at the fifteen year horizon to produce a significant improvement in the human interface for computer-based problem solving. A key component of this program is devoted to the development of intelligent terminal capabilities.

Proposed research approach. A four step approach is recommended for developing WWMCCS relevant intelligent terminal technology.

1. Develop a base system for experimentation,
2. develop terminal resident components of the larger data management system in parallel with
3. investigations of the man/machine interface, and
4. produce a prototype intelligent terminal that combines terminal resident data management capabilities with an improved human interface to those capabilities.

The development of an experimental vehicle will require hardware and software subtasks to acquire and construct the physical unit and to design, program, and test a flexible system for experimentation. The hardware used for the preliminary research study is all on short term loan and is not available for a long term program. The design of the experiment support software will be based on the experience gained during the preliminary research study.

The first approach to the development of terminal resident data management system components will be to model, in a coarse way, the terminal, network, and host environment. This coarse model will be used to determine which activities in the data management system are best suited to distributed data management systems accessed via intelligent terminals. There are major constraints on intelligent terminal activities.

1. Only a limited amount of secondary storage is available at the terminal.
2. The limited CPU power of a minicomputer restricts the type of functions that may be rapidly and efficiently processed in the terminal.
3. The use of dialed phone lines constrains the bandwidth between the terminal and a large host.

Those DMS components offering the most promise for an intelligent terminal environment will be designed and implemented on the experimental terminal system. The implementation of terminal resident software will require enhancements in the large host oriented experimental data management system to facilitate cooperation between the two activities. Some obvious potential study areas include:

1. paging strategies for management of intelligent terminal as a cache to the large host data management system,
2. indexing strategies (e.g., it might be advisable to do some of the host DMS index processing at the terminal or to provide network-wide distributed data base indices),
3. automated backup, failure detection, and recovery,
4. terminal completion of host DMS processing (e.g., the host might send a compressed report to the terminal for final

formatting, or the host might send several records which may or may not satisfy the user's query to the terminal for final sifting), and

5. multiple host applications where the parallelism of the multiple host environment is used to speed up processing.

The approach in the man/machine interface area is to use the experimental terminal to study various interface concepts and design alternatives. At a minimum, the touch and display capabilities would be exploited. Automatic call-out units would permit the terminal to handle all telephoning chores, to re-dial hosts when connections are broken, etc. Audio output, color capabilities, etc., could be added as needed for further experiments. Since the group will be working on the same hardware base used to develop data management capabilities, it is only natural and, of course, desirable that the choice of interface demonstrations will revolve around distributed data management appli-
demonstrations will revolve around distributed data management applications. Some areas of immediate interest that might be explored are listed below.

1. Network transparency can include, for example, an ability for the terminal to automatically dial into network front-ends, issue the appropriate command language to connect to a host, issue the host commands to invoke an application package, handle line idiosyncrasies and recover from failures. Transparency may also include the location of network resources, regardless of where they are. On a heterogeneous network facilities may be included so that the user does not have to know the log-in sequences, command languages, etc., of the various hosts which support his application.
2. The use of touch/display data management command languages may be both integrated into the experimental distributed data management resources system as a basic capability and investigated on a retrofit basis for a large-host system like WWDMS.
3. Automated user assistance may include multi-screen help displays and special scratch pad displays.

4. Data compression of reports, graphics and dot-density pictures, for the improvement of communication line bandwidth, may be performed in the terminal.
5. Ciphers may be implemented in the terminal software to investigate software based end-to-end (terminal to host) and hop-by-hop (terminal to NFE), encryption for highly secure applications.

Over the three year research activity, the original experimental system should evolve, in both hardware and software, into a prototype intelligent terminal which can demonstrate a superior interface to a sophisticated data management system, parts of which will actually reside in the terminal.

Anticipated results. At the end of fiscal 76, the basic intelligent terminal should be assembled and available for limited demonstration. Early in fiscal 77 some preliminary components of the data management system should be operational within the intelligent terminal. Late in the second year of the program, it is anticipated that sufficient experience will have been gained with several man/machine interface approaches so that their appropriateness or inappropriateness for exploitation in the WIN can be assessed. Near the end of the three year program, a prototype intelligent terminal should be available which demonstrates terminal capabilities in the area of cost, performance, survivability, and the human interface for WWMCCS applications. The experimental prototype may not be directly exportable to the PWIN or WIN. More likely, it will require modification or re-implementation to meet production PWIN and WIN requirements. Throughout the three year program, the intelligent terminal should be available for occasional demonstrations at WWMCCS sites. This will apprise potential users of the evolving data management and terminal technologies and provide them with an opportunity to make input to the research program.

## Areas Not Included in This Research Plan

Introduction. There are some areas of research which, although important, we do not plan to address. In some cases, there is little real research involved, but only a time-consuming study of known alternatives, coupled with an implementation effort. In other cases, real, high-risk research is needed. However, the CAC does not have the manpower or expertise to cover the whole field of data management. We have been obliged to omit some areas which we hope will be satisfactorily covered by other research groups. Since the research in some of these areas is coupled to the research we plan to undertake, we anticipate that some significant portion of staff time will be spent studying those areas and maintaining liaison with other research groups. In this section we briefly describe the omitted areas, together with the impact that not solving these problems will have on the overall research and development effort.

Project management schemes. The management of a network installation or of projects using several network hosts requires novel management tools. The areas of accounting, billing, and resource allocation to individuals by project managers are of prime interest. There are a number of technical problems involved in maintaining a distributed accounting and resource control base and assuring its security. Several of these problems are actually being addressed in this research plan, but under different headings. For example, the problem of updating and maintaining distributed accounting files is only one application of the general problem of updating and maintaining a distributed data base. Areas like project administration have been well addressed in systems like Multics and need the intelligent extension of proven techniques to the heterogeneous network environment.

Network monitoring and accounting. Frequent (and for some aspects, continual) monitoring of the network must be carried out to assure users that the network is working properly and to identify potential trouble areas or bottlenecks. In addition to monitoring and evaluating network performance, a centralized network control center (such as exists in the ARPA network) should be responsible for protocol evaluation (see discussion of protocol research areas, below) and for certifying that the various sites have correctly implemented those protocols. These monitoring and certification functions play an important role in the reliable operation of any network. The problems, however, are largely ones of engineering and implementation, although a better understanding of performance evaluation techniques would clearly be helpful.

Integrity. Integrity refers to the problem of maintaining the accuracy and completeness of the information in a data base. At the present time, the principal technique for identifying loss of integrity is consistency checking. This usually means devising a set of ad hoc rules, specific to the particular data base, to check the data. This area is in serious need of work to determine whether some more systematic approach is possible. A systematic approach could then be built into the automatic, self-tuning data management system. It could well be that no systematic approach is feasible; in this case consistency checks will continue to be generated tediously by hand.

Once it has been determined that data has been lost or is in error, the usual scheme is to appeal to a backup copy for the correct data. The development of a more systematic error detection and recovery scheme is, at best, high risk and may be impossible.

Protocol research areas. In a network environment, the problem of accessing resources is totally different from what it is in the traditional computer system, in part because of the absence of a common store. To solve this problem, protocols are developed to allow the communicating computers to have a common basis for interaction and control. These protocols are central to accomplishing any sort of distributed computing or load sharing. In addition, the success of any distributed computing effort is dependent on the capability of the protocols involved to provide the required functions efficiently and reliably.

There are several aspects of protocols that require further investigation. Among these is the development of a methodology for defining protocols that will help to avoid ambiguities that may arise when different groups attempt to implement these protocols. More work is also needed to characterize how a protocol and its implementation may be made resilient to failures or abuse. This is a very difficult problem; it is hard even to define rigorously what one means by "resilient". In addition, there are the specific problems of arriving at a framework that efficiently and flexibly expresses the operations of the protocol for a diverse group of computers. For example, a file transfer protocol must be able to accommodate many file formats.

In summary, a major research effort, essential to the correct functioning of a distributed system in a network, is required in these areas.

Front-end research areas. It is clear from ARPA and PWIN experiments that network control software consumes significant host resources and is difficult to design and implement. One concept is to provide a network front end (NFE) computer that handles the bulk of network control functions and maps them into a form more palatable to the host computer. The NFE's envisioned usually connect a host to the network and also handle the local terminal load. While simplified protocols for host to front-end communication have been proposed, none has yet been accepted as a standard. There are no working examples of an NFE and attempts to produce one have, so far, been unsuccessful. Furthermore, researchers do not know how much load and what kinds of functions can be taken off of a host and transferred to an NFE. NFE's seem to promise improved reliability, better network user access, and reduced load; but the achievement of that promise is yet to come.

The other front-end area of concern is the secure front-end processor (SFEP), a security controller being developed for PWIN. Since adequate security support is critical to the future of WWMCCS networking, successful development of the SFEP is essential.

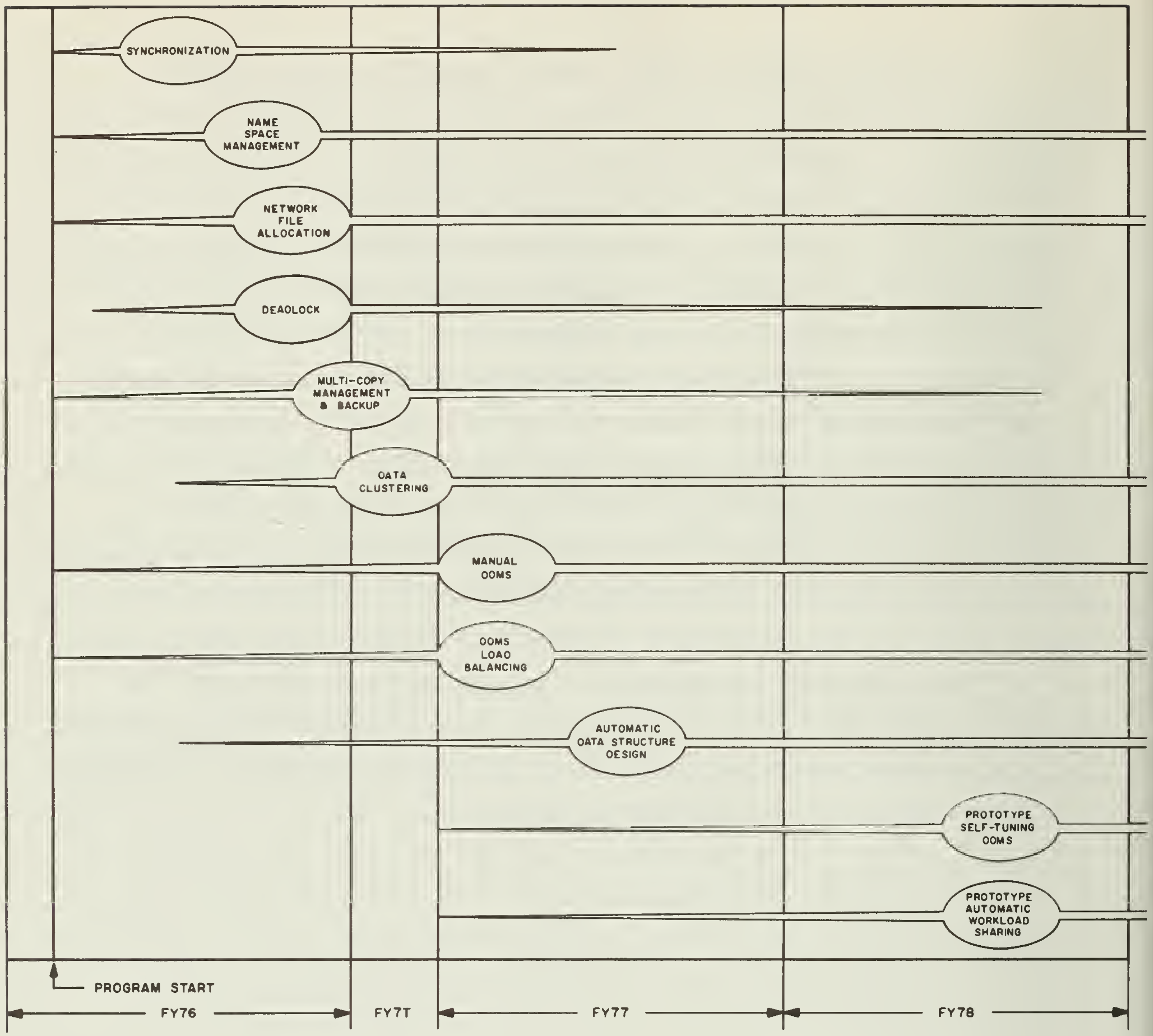



FIGURE 3
PHASING OF RESEARCH AREAS IN BASIC PROGRAM

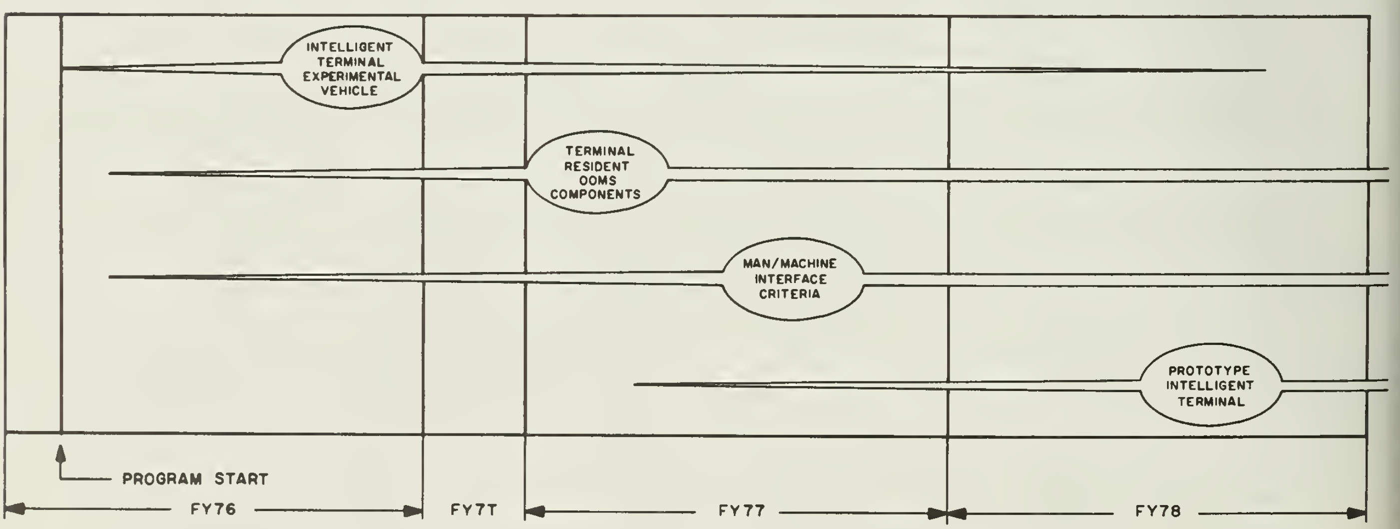



FIGURE 4
PHASING OF RESEARCH AREAS IN OPTIONAL INTELLIGENT TERMINAL PROGRAM

# Management Considerations

## Schedule

Due to the close interaction between research areas, it is inappropriate to consider spearately manned and scheduled activities for each of the 15 research areas addressed. Instead, the individual areas are addressed in an integrated fashion within the basic modeling and experimental activities.

Figure 3 shows the times at which significant results are anticipated in each of the 11 basic data management and resource sharing areas. Unanticipated difficulties and breakthroughs are likely and should be expected to occasionally shift these events forward or backward in time. For each area, the approximate beginning and termination of significant work are implied by the leaders and trailers.

Figure 4 shows the additional events (4 research areas) that are anticipated if the intelligent terminal option is selected.

## Personnel Requirements

Overview. Research personnel should consist of a principal investigator (PI) responsible for the technical quality of the whole program, senior investigators (SI) responsible for the modeling and experimental components and systems analysts (SA) and graduate research assistants (GRA) as staff support. The optional intelligent terminal program will require one or more hardware engineers (HE) and a human factors (HF) expert (e.g., an industrial engineer) if one of the larger program alternatives is undertaken.

Skill levels. Senior investigators have major management responsibilities and must be competent, creative researchers. System analysts will be required to participate in modeling and experimental activities. Broad competence in the theoretical aspects of computer networks and in the design and implementation of software systems is expected. Strong expertise in at least one theoretical or empirical area is necessary. Graduate research assistants should be candidates for a Master or Ph.D. degree in computer science.

Manpower loading. Tables 3 and 4 summarize the personnel requirements for FY76 through FY78. Estimated man-loadings and man-months are shown for several alternatives. Tables 5 and 6 show the total man-month requirements for various combinations of alternatives.

| Program Alternatives | Major Components | Skill Level - Man Loading | | | | | | | Man-Months |
|---|---|---|---|---|---|---|---|---|---|
| | | PI | SI | SA | GRA | HE | HF | Total | |
| Minimum basic data management program | Modeling | ½ | 1 | 1½ | ½ | | | 3½ | 47 |
| | Experimental system | ½ | 1 | 1½ | 1 | | | 4 | 53 |
| | Technology transfer | | | ½ | | | | ½ | 7 |
| | Total | 1 | 2 | 3½ | 1½ | | | 8 | 107 |
| Moderate basic program | Modeling | ½ | 1 | 2 | 1 | | | 4½ | 60 |
| | Experimental system | ½ | 1 | 2½ | 1½ | | | 5½ | 73 |
| | Technology transfer | | | 1 | ½ | | | 1½ | 20 |
| | Total | 1 | 2 | 5½ | 3 | | | 11½ | 153 |
| Minimum option | Intelligent terminal | | | | 1 | ¼ | | 1¼ | 17 |
| Expanded option | Intelligent terminal | | 1 | 2 | 1 | 1 | 1 | 6 | 80 |

Table 3

Personnel Requirements for FY76 plus FY7T
(13.33 months beginning 21 August 1975)

| Program Alternatives | Major Components | Skill Level - Man Loading | | | | | | | Man-Months |
|---|---|---|---|---|---|---|---|---|---|
| | | PI | SI | SA | GRA | HE | HF | Total | |
| Basic program with emphasis on dynamic restructuring | Modeling | ½ | 1 | 1½ | 1 | | | 4 | 48 |
| | Experimental system | ½ | 1 | 3 | 1 | | | 5½ | 66 |
| | Technology transfer | | | ½ | | | | ½ | 6 |
| | Total | 1 | 2 | 5 | 2 | | | 10 | 120 |
| Basic program with emphasis on load sharing | Modeling | ½ | 1 | 1½ | 1 | | | 4 | 48 |
| | Experimental system | ½ | 1 | 4 | 2 | | | 7½ | 90 |
| | Technology transfer | | | ½ | | | | ½ | 6 |
| | Total | 1 | 2 | 6 | 3 | | | 12 | 144 |
| Expanded program for both dynamic restructuring and load sharing | Modeling | ½ | 1 | 2½ | 1½ | | | 5½ | 66 |
| | Experimental system | ½ | 2 | 6 | 2 | | | 10½ | 126 |
| | Technology transfer | | | 1 | ½ | | | 1½ | 18 |
| | Total | 1 | 3 | 9½ | 4 | | | 17½ | 210 |
| Basic option | Intelligent terminal | | | 1½ | 1 | ½ | | 3 | 36 |
| Expanded option | Intelligent terminal | | 1 | 2 | 1 | 1 | 1 | 6 | 72 |

Table 4

Personnel Requirements for FY77
(FY78 requirements are identical)

| | no intelligent terminal | minimum intelligent terminal | expanded intelligent terminal |
|---|---|---|---|
| minimum basic | 107 | 124 | 187 |
| moderate basic | 153 | 170 | 233 |

Table 5

Total Man-Months for FY76 plus FY7T Alternatives

| | no intelligent terminal | basic intelligent terminal | expanded intelligent terminal |
|---|---|---|---|
| basic restructuring | 120 | 156 | 192 |
| basic load sharing | 144 | 180 | 216 |
| expanded restructuring and load sharing | 210 | 246 | 282 |

Table 6

Total Man-Months for FY77 or FY78 Alternatives

## Computer Resource Requirements

Computer networks. Continuing access to the ARPA network is required. Occasional access to the PWIN is desirable. The ARPA network should be the primary network resource due to its similarity to the PWIN and its 24-hour availability for research purposes.

Network host computers. While heterogeneous experiments will be performed, the Multics facilities will support much of the initial development and basic capabilities of the experimental systems. The Multics hosts on the ARPA network offer good network support and can run an encapsulated GCOS system. Access to at least MIT's Multics facility and RADC's Multics facility is required for multi-host experiments. Occasionally, the Multics facility at Honeywell's Cambridge Information Systems Laboratory is available on the ARPA network and there is serious consideration being given to adding Honeywell's Phoenix system to the network. Access to the Honeywell systems would also be desirable.

Intelligent terminal hardware. Hardware requirements for an intelligent terminal program vary widely. A minimum system, similar to that used for the preliminary research study, which preceeded this plan, would cost approximately $30,000 per copy.

Minimum system

| | |
|---|---|
| PDP-11/10 CPU and 24K wds | $12,000 |
| high speed plasma panel with interface and touch input | 6,500 |
| floppy disk | 4,500 |
| modems, keyboard, spare parts, etc. | 7,000 |
| | $30,000 |

A sophisticated system similar to that which will be economically feasible in the early 1980's would cost approximately $87,000 per copy.

Sophisticated system

| | |
|---|---|
| PDP-11/45 CPU and 64K wds | $44,000 |
| 3 touch/display panels* | 19,500 |
| floppy disk | 4,500 |
| modems, keyboard, auto call unit, experimental input/output devices | 9,000 |
| racks, power supplies, spares, etc. | 10,000 |
| | $87,000 |

* color CRT technology may be substituted

Thus the price of intelligent terminal hardware ranges from $30,000 through $90,000 per copy depending on the degree of sophistication. The research program would require at least one copy in addition to any copies desired by the WWMCCS community for test and evaluation purposes.

Deliverables

Quarterly and event-driven deliverables are recommended. Quarterly technical progress reviews will provide JTSA with an opportunity to monitor progress and to input direction and guidance to the program. In addition, tutorials on recent research results should be presented during quarterly reviews. Event-driven deliverables will be provided as mutually agreed by JTSA and the CAC. Event-driven deliverables will include working papers, technical reports, technical assistance to JTSA, consultation to WWMCCS end-users, and demonstrations as appropriate. Table 7 summarizes the deliverables.

| Quarterly Deliverables |
| --- |
| technical progress review<br>tutorials on recent research |
| Event-Driven Deliverables |
| working papers<br>technical reports<br>technical assistance to JTSA<br>consultation to WWMCCS end-users<br>demonstrations<br>manual DDMS<br>prototype self-tuning DDMS<br>prototype automatic WLS system<br>prototype intelligent terminal<br>other demonstrations as appropriate |

Table 7

Deliverables

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>CAC Document Number 164<br>JTSA Document Number 5510 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>Research in Network Data Management and Resource Sharing - Research Plan | | 5. TYPE OF REPORT & PERIOD COVERED<br>Research Report - Interim |
| | | 6. PERFORMING ORG. REPORT NUMBER<br>CAC Document Number 164 |
| 7. AUTHOR(s) | | 8. CONTRACT OR GRANT NUMBER(s)<br>DCA100-75-C-0021 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>Center for Advanced Computation<br>University of Illinois at Urbana-Champaign<br>Urbana, Illinois 61801 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Joint Technical Support Activity (DCA)<br>11440 Isaac Newton Square, North<br>Reston, VA 22090 | | 12. REPORT DATE<br>June 19, 1975 |
| | | 13. NUMBER OF PAGES<br>62 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) | | 15. SECURITY CLASS. (of this report)<br>UNCLASSIFIED |
| | | 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

Copies may be obtained from the address given in (11) above.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

No restriction on distribution

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Distributed data management
Resource sharing
Network data management

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A three year research program in distributed data management and resource sharing is presented.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)