

A major purpose of the Technical Information Center is to provide the broadest dissemination possible of information contained in DOE's Research and Development Reports to business, industry, the academic community, and federal, state and local governments.

Although portions of this report are not reproducible, it is being made available in microfiche to facilitate the availability of those parts of the document which are legible.

**3**

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36

LA-UR--87-2241

DE87 013133

TITLE: METHODOLOGY FOR MATERIALS CONTROL AND ACCOUNTING INFORMATION SYSTEMS

AUTHOR(S): P. Helman and R. B. Strittmatter

SUBMITTED TO 28th Annual Meeting of the Institute of Nuclear Materials Management, Los Angeles, July 12-15, 1987

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof

By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution or to allow others to do so, for U.S. Government purposes

The Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy

MASTER

Los Alamos Los Alamos National Laboratory Los Alamos, New Mexico 87545



## METHODOLOGY FOR MATERIALS CONTROL AND ACCOUNTING INFORMATION SYSTEMS

P. Helman, University of New Mexico, Dept. of Computer Science, Albuquerque, NM

R. B. Strittmatter, Los Alamos National Laboratory, Los Alamos, NM

### ABSTRACT

Modern approaches to nuclear materials safeguards have significantly increased the data processing needs of safeguards information systems. Implementing these approaches will require developing efficient, cost-effective designs. Guided by database design research, we are developing a design methodology for distributed materials control and accounting (MC&A) information systems. The methodology considers four design parameters: network topology, allocation of data to nodes, high-level global processing strategy, and local file structures to optimize system performance. Characteristics of system performance that are optimized are response time for an operation, timeliness of data, validity of data, and reliability. The ultimate goal of the research is to develop a comprehensive computerized design tool specifically tailored to the design of MC&A systems.

### 1. OVERVIEW OF MATERIALS CONTROL AND ACCOUNTING (MC&A) SYSTEMS

Computerized capture of data used for materials accounting, analysis of safeguards information, and the preparation of reports is becoming routine throughout the safeguards community. The informational requirement related to the safeguards objective is to keep timely information on the location and status of nuclear materials and on the personnel with access to these materials. This information can aid in the detection of unauthorized actions resulting in the theft or diversion of nuclear material and alternatively provide positive assurance that no theft or diversion of nuclear material has occurred. As existing safeguards systems are expanded to meet the evolving safeguards requirements and as new systems are developed, significantly increased demands are placed on the information systems. The safeguards information systems are required to handle increased quantities of data with increased analysis of safeguards information.<sup>1</sup> The integration of physical security, materials control, and materials accounting components of the safeguards

system and the use of information not traditionally associated with safeguards activities, including process control and health and safety information, will place additional demands on the safeguards computer systems.<sup>2</sup> Increased awareness of the insider threat has placed additional requirements on access control for the information system and the separation of data access functions across safeguards boundaries. To implement the emerging information-intensive integrated safeguards systems will require developing efficient, cost-effective information systems that minimize the impact on process operations. Distributed-processing hardware and software have the potential of greatly enhancing transaction-based systems, both from the safeguards and security effectiveness perspective and from the perspective of impact on process operations.<sup>3</sup> This paper summarizes a methodology for performing distributed database design optimization applied to materials control and accounting information systems.

### 2. A COMPUTERIZED DESIGN METHODOLOGY: AN OVERVIEW

The design of a distributed database system for an MC&A application (or any similar application) requires careful consideration of several design parameters. To focus on a specific application of this methodology, we will assume that the basic unit for allocation of data is a materials balance area (MBA). The methodology can, however, be applied to any distributed processing system with well-defined transactions. Of primary importance are the following parameters.

1. Network Topology. That is, how should the computers and terminals be spread out through the plant and how should they be interconnected. Network topology includes the issue of how much computing power should be placed at each node of the network. For example, in an extremely centralized design, only terminals are placed at each MBA of the plant, and each of these terminals is connected to a central computer. In contrast, in an extremely decentralized design, a collection of homogeneous computers are distributed to the MBAs of the plant with high-speed communication links connecting them together. The design process must find the compromise between these extremes that is optimal for a given collection of processing and security requirements.

\*This work supported by the U.S. Department of Energy, Office of Safeguards and Security.

2. Allocation of Data to Nodes. Once a network topology is selected, an important design decision is how to allocate the data to the nodes of the network. Extreme strategies include storing all the data at a single node of the network, partitioning the data (without replication) among the nodes of the network, and duplicating all the data at each node of the network. The design process must find the compromise between these extremes that is optimal for a given collection of processing and data allocation requirements.
3. High-Level Global Processing Strategy. This includes how complex information processing operations should be decomposed into smaller operations, how update operations should be batched, and how processing and data should be routed through the network.
4. Local File Structures. Once design issues 1-3 have been settled, low-level file design must be performed for each node of the network. File design includes record definitions, record clusterings, and file indexing.

Our methodology considers each of these four design parameters in an attempt to optimize system performance. Specifically, the methodology optimizes the following characteristics of system performance.

Response Time For An Operation: The amount of real time that elapses between the initiation and completion of an operation.

Timeliness of Data: How much out of date the data is that an operation uses to perform its task and produces as output. Note that some operations will be defined so that the data they use must be completely current, implying that unprocessed updates will have to be applied before these operations can be run.

Validity of Data: The error rate for the data that an operation uses to perform its task and produces as output. Invalid data could result from posting a transaction that is not valid or from faulty network communications.

Reliability: The probability that an operation will be able to run when requested. This probability is determined by computer and communication link failure rates.

The ultimate goal of our research is to develop a comprehensive, computerized design tool specifically tailored to the design of MC&A systems. The key components of the design tool include:

Flexible Requirements Specification. The computerized design tool must be applicable to the design of MC&A systems in general, rather than to the design of a single MC&A system. Consequently, the details of the information processing tasks and data allocation constraints must be allowed

to vary for each system being designed. As is described in Section 3.1, our tool provides a convenient framework for stating the requirements of each specific MC&A application under study.

Integrated Design Optimization. The methodology addresses each of the four design parameters mentioned above.

Multi-Level Analytical Cost Evaluation and Optimization. At the heart of the design methodology is an analytical model of cost. This model attempts to predict how well each candidate design will support the requirements of the given MC&A application. As is described in Section 3.2, the cost model has three evaluation levels:

The Level I evaluation which is used to compare different data allocation schemes and to compare, at a gross level, different network topologies.

The Level II evaluation which is used for a detailed comparison of network topologies and processing strategies.

The Level III evaluation which is used to compare local file designs.

The design tool is being implemented so that these evaluators can be applied collectively to optimize an entire MC&A system "from scratch" or independently to optimize any single component of an MC&A system. In Section 4 we describe computerized algorithms for optimizing MC&A systems with respect to the Level I evaluation. We also indicate current research directions for optimizing MC&A systems with respect to the Level II evaluation. Reference 4 describes computerized algorithms for the Level III optimization.

Sensitivity Analysis Components. The design tool is being implemented so as to include sensitivity analysis components that help an analyst assess the effects on system performance of certain types of data allocation constraints. For example, the tool can be used to compare, for a given set of MC&A processing requirements, the optimal system design obeying a specified collection of data allocation constraints and the optimal design possible if some of those constraints were relaxed. Section 4 provides the details.

### 3. DETAILS OF THE METHODOLOGY

The experience of many database researchers shows that detailed models of computer networks quickly become too complex to be of practical use. Consequently, one of our major research tasks was to arrive at the proper level of abstraction for the model. We attempt to include in the model only those details of the requirements of an MC&A application and of a system design that are most critical to the predication of performance characteristics. We believe that this abstraction is necessary, even at the expense of some precision, in order to produce a usable design tool.

### 3.1 Requirement Specifications

We have attempted to keep manageable the amount of detail present in the requirement specifications that define an MC&A application. This is necessary to address concerns for computational tractability as well as concerns for what information we realistically can expect to obtain from the future users of the MC&A system under construction.

The following collection of information requirements is at a high-level of abstraction, yet is sufficient for optimizing the design with respect to the Level I and Level II evaluators. We note, however, that when we later perform local file design optimization with respect to the Level-III evaluator, more detailed input will be required.<sup>4</sup>

The requirement specifications consists of four components.

Plant Description. This is simply the number of MBAs in the plant. Each MBA will be referred to as a node.

Data Fragments. These are the basic units of data allocation; a data allocation scheme allocates each of the specified data fragments to one or more nodes of the computer network. We anticipate that the collection of data fragments will differ little between MC&A applications. An example of a data fragment is "the collection of Book Inventory records that pertain to inventory items currently in MBA X." The reader is referred to Reference 5 for a good discussion on how to define data fragments in the general context of distributed database design.

Forbidden Fragment-to-Site Assignments. These are the data allocation constraints for the system. These constraints model attempts to limit access to portions of the database to provide increased protection against the insider threat. Each constraint is of the form (DF, N) and specifies that data fragment DF cannot be allocated to node N.

Operation Templates. The operation templates characterize the information processing that must be supported by the MC&A application. Each operation template consists of the following components.

**FREQUENCY:** The relative frequency with which the operation is performed.

**INITIATION:** The node from which the operation is initiated.

**DATA:** The data fragments required by the operation.

**PROCESSING:** A quantitative description of the intensity and type of processing that must be performed.

For example, the following illustrates these four components of the template for the hypothetical operation "Transaction Entry and Simple Validation."

INITIATION: MBA X

FREQUENCY: One operation per t time units

DATA:

Retrieve Only

Fragment: Book Inventory Data with (MBA = X)  
Quantity: 50 bytes

Fragment: Validation Data with (MBA = X)  
Quantity: 100 bytes

Update

Fragment: Transaction History Data with  
(MBA = X)  
Quantity: 20 bytes

PROCESSING: Category I (low intensity)

Also part of an operation template is an evaluation vector that specifies the criteria by which we are to evaluate how well a particular operation is performed. An evaluation vector for operation  $O_k$  is of the form

$$\langle RSP_k, TML_k, VLD_k, RLB_k \rangle,$$

where

RSP is a function that specifies the value of fast response time.

$RSP_k(\tau) = v$  means that response to  $O_k$  in  $\tau$  units is worth  $v$ .

TML is a function that specifies the value of timely data.

$TML_k(d) = v$  means that using data that is  $d$  units out-of-date in performing  $O_k$  is worth  $v$ .  
Note: If  $O_k$  must be performed with current data, define  $TML_k(d) = -\infty$  for  $d > 0$ . This makes full timeliness an absolute constraint for  $O_k$ .

VLD<sub>k</sub> is a constant that specifies how disastrous performing  $O_k$  with at least some invalid data would be. VLD<sub>k</sub> typically will be negative for all  $O_k$ .

RLB is a function that specifies the criticality of performance.  $RLB_k(p) = v$  means that being able to perform  $O_k$  with probability  $p$  is worth  $v$ . Note that this probability is determined by what data fragments  $O_k$  needs to access, where those data fragments are allocated, and network failure probabilities.

Observe that the framework of an evaluation vector allows certain operations to evaluate well under a design that, for example, provides good response time, whereas other operations are indifferent to this system performance characteristic. Thus, the framework allows a system design

to be evaluated with respect to the needs of each particular application.

The following is an evaluation vector for the hypothetical operation "Transaction Entry and Simple Validation" (the values chosen for the evaluation functions are only illustrative):

RSP:  $RSP(t) = 100$ , for  $0 < t < 5$   
 $RSP(t) = 0$ , for  $5 < t < 25$   
 $RSP(t) = -100$ , for  $t > 25$

An RSP function such as this indicates that fast response time is very desirable and response time  $>25$  units is unacceptable.

TML:  $TML(0) = 0$   
 $TML(d) = -\infty$ , for  $d > 0$

A TML function such as this indicates that all data used by the transaction must be made current before the transaction can execute.

ILD:  $-20$

Such a value might indicate that using erroneous book inventory values to construct the transaction is certainly undesirable, though not catastrophic.

RLB:  $RLB(p) = 100 * p$

An RLB function such as this indicates that the reliability benefit increases linearly with the probability of performance.

## 1.2. Three Levels of Evaluation and Optimization

Our cost model is guided by widely accepted research in database design demonstrating that overall system performance is most directly impacted by the following five factors:

- 1) amount of data and number of messages transmitted from node to node,
- 2) amount of data and number of blocks transferred from disk to processor,
- 3) intensity of required processing,
- 4) delay due to lockout and network backlog,
- 5) processing power of each computer in the network.

Our mathematical model of these performance factors is partitioned into three levels:

Level I: The Level I evaluation is applied to a given network topology and data allocation scheme for the network. The evaluation is a simple function of the amount of data transmission required to process the set of operations and of a simple model of system reliability. Intuitively, whenever an operation that must read  $b$  bytes of data fragment  $f$  is initiated at a node that does not contain a copy of fragment  $f$ ,  $b$  bytes of fragment  $f$  must be transmitted. Whenever an operation  $O$  updates  $b$  bytes of fragment  $f$ ,  $b$  bytes of data must be transmitted to each node (other than  $O$ 's

initiation node) that contains a copy of  $f$ . System reliability is modeled by a simple estimate of the probability that at least one node containing required data is operational. Section 4 contains the mathematical statement of the Level-I evaluation and presents an optimization algorithm for finding the data allocation scheme that minimizes the Level-I evaluation for a given network.

Level II: Although a Level-I evaluation is sufficient for comparing data allocation schemes for fixed network topologies, a more detailed measure of system performance is required to compare different topologies. These measures, which are based on the operations' evaluation vectors and processing times (including network delays), yield an expectation of system performance. Current research is attempting to correlate processing strategies against fixed network topologies with processing times and to develop algorithms for constructing optimal, or near optimal, topologies.

Level III: The Level-III evaluation is used to compare local file structure schemes for each node of the selected network. This evaluation is based on the number of disk accesses required to process the operations at a local node. The evaluation requires a more detailed specification of the required operations than is supplied in the templates (e.g., what attributes must be retrieved, what attributes are qualified in the retrieval). Reference 4 presents algorithms for optimally structuring files with respect to the Level-III evaluation.

Although these evaluators can be applied independently to optimize the various components of a design, our methodology integrates the evaluators and optimization algorithms into a package for performing a comprehensive design. This integration is extremely desirable because the design components are highly interrelated. Notice, for example, that the first step in evaluating a given network topology is to find an optimal allocation scheme for it. Only after the allocation scheme has been selected can we perform a Level-II evaluation in which different processing strategies are considered and network delays are accounted for.

## 4. THE LEVEL-I EVALUATOR AND AN OPTIMAL ALLOCATION ALGORITHM

The remainder of this paper focuses on the data allocation problem.

### 4.1. Mathematical Statement of the Level-I Evaluator

Let  $A$  be an allocation scheme for a given network topology. The cost of  $A$  is the difference

$$\text{Cost}(A) = \text{TRM}(A) - \text{RLB}(A) .$$

The TRM component of Cost is the number of bytes of data (per time unit) that must be transmitted

to process the operations against allocation scheme A. The RLB component measures the reliability provided by allocation scheme A (the higher the RLB component the better). This quantity is based on the probability that at least one node containing required data is operational.

The statement of the TRM component of the cost function uses the following notation.

- $A(f_i)$  - set of nodes to which allocation scheme A allocates a copy of  $f_i$
- $D(n,S) = 0$  if  $n \notin S$ ; 1 if  $n \in S$
- $I(O_k)$  - the node from which operation  $O_k$  is initiated
- $r_k$  - the retrieve portion of operation  $O_k$
- $u_k$  - the update portion of operation  $O_k$
- $AMT(r_k, f_i)$  - number of bytes of  $f_i$  required by  $r_k$
- $AMT(u_k, f_i)$  - number of bytes of  $f_i$  updated by  $u_k$
- $C$  - a constant that scales the difference between retrieval and update costs

TRM(A) is defined as

$$TRM(A) = \sum_{k=1}^q \left\{ \sum_{i=1}^d \text{Freq}(O_k) * RSP_k * D[I(O_k), A(f_i)] * AMT(r_k, f_i) \right\} + C * \sum_{k=1}^q \left\{ \sum_{i=1}^d \text{Freq}(O_k) * RSP_k * |A(f_i) - I(O_k)| * AMT(u_k, f_i) \right\} .$$

The RLB component of the cost function is based on the following quantity introduced in Ref. 5.

$$B(c, f_i) = (1 - 2^{1-c}) * D_i .$$

$B(c, f_i)$  is an estimator of the reliability value of allocating  $c$  copies of fragment  $f_i$ .  $D_i$  is meant to reflect the value of having fragment  $f_i$  always available. For example,  $D_i$  could be derived from the RLB components of the operation templates as

$$D_i = \sum_{O_k \text{ retrieves } f_i} RLB_k .$$

The reliability value of allocation scheme A is then defined as

$$RLB(A) = \sum_{i=1}^d B[|A(f_i)|, f_i] .$$

#### 4.2. The Algorithm Alloc

We observe that when no fragment-to-node constraints are present, the number of allocation schemes is  $(2^m - 1)^d$ , where  $m$  is the number of nodes in the network and  $d$  is the number of data fragments. Consequently, solution to the optimization problem by exhaustive search is infeasible, even for moderate size problem instances. We now present a very efficient algorithm for constructing an optimal data allocation scheme.

The following "differential" quantities are central to the optimization algorithm.

Let  $f$  be any fragment,  $n$  any node, and  $D'(n,S) = [1 - D(n,S)]$ . Define

$$TRM\text{-Save}(f,n) = \sum_{k=1}^q \text{Freq}(O_k) * RSP_k * D'[I(O_k), (n)] * AMT(r_k, f) - C * \sum_{k=1}^q \text{Freq}(O_k) * RSP_k * D[I(O_k), (n)] * AMT(u_k, f) .$$

$$RLB\text{-Gain}(c,f) = B(c,f) - B(c-1,f) .$$

The optimal allocation algorithm is as follows.

#### Algorithm Alloc:

for each fragment  $f_i$  do

(\* Pre-process\*)

for each node  $n_j$  such that  $(f_i, n_j)$  is not forbidden do

Compute  $V_j = TRM\text{-Save}(f_i, n_j)$  and place  $(n_j, V_j)$  on list L

(\* Every fragment has to be allocated to at least one node \*)

Allocate a copy of  $f_i$  to a node  $n_j$  with maximal  $V_j$  value.

Remove from list L  $(n_j, V_j)$

$c := 1$

Improve := TRUE

while (Improve and (there remain nodes on L to consider) do

Let  $n_j$  a node on L with maximal  $V_j$  value

if  $[V_j + RLB\text{-Gain}(C+1, f_i)] > 0$

then allocate a copy of  $f_i$  to  $n_j$

Remove from list L  $(n_j, V_j)$

$c := c + 1$

else Improve := FALSE

end while

end for each fragment  $f_i$

It has been proven that this algorithm is guaranteed to find an optimal allocation scheme, and an  $O(m * d * \max\{q, \log m\})$  time implementation is presented ( $m$  is the number of nodes in the network,  $d$  is the number of data fragments, and  $q$  is the number of operations).<sup>6</sup>

#### 4.3. Sensitivity Analysis

Once algorithm Alloc terminates with an optimal allocation scheme, the analyst is placed in an interactive sensitivity analysis mode. From this mode, the analyst is prompted for modifications to the collection of forbidden fragment-to-node assignments. The analyst can add one or more forbidden pair constraints or he can delete

one or more existing forbidden pair constraints. The algorithm will construct a new allocation scheme that is optimal with respect to the new collection of constraints, compare the cost of this new optimal scheme with the cost of the old scheme, and then give the analyst the opportunity to repeat this process as often as desired. Thus, the sensitivity analysis component is a very convenient tool for assessing the effects on system cost of proposed data allocation constraints.

Following are high-level descriptions of the algorithms for adding and deleting forbidden pair constraints (f,n).

```
AddPair(f,n,c)
(* Add the forbidden pair constraint (f,n). A
is an optimal allocation scheme before the for-
bidden pair constraint (f,n) is added and A
allocates c copies of fragment f. *)
```

```
  if (A allocates a copy of fragment f to node
n)
  then
    De-Allocate fragment f from node n

    Let m be a node with maximal
    TRM-Save(f,.) value among those nodes
    to which A does not allocate a copy of
    fragment f and (f,n) is not a forbidden
    pair

    if {(c=1) or
        {TRM-Save(f,m) + RLB-Gain(c,f) > 0}}
    then Allocate fragment f to node m
    else c := c-1
```

```
  end if
```

```
end AddPair
```

```
DeletePair(f,n,c)
(* Remove the forbidden pair constraint (f,n).
A is an optimal allocation scheme before the
forbidden pair constraint (f,n) is relaxed and
A allocates c copies of fragment f. *)
```

```
  Let m be a node with minimal TRM-Save(f, *)
  value among those nodes to which A allocates
  a copy of fragment f
```

```
  if [TRM-Save(f,n) > 0] or [TRM-Save(f,n) >
TRM-Save(f,m)]
  then
    Allocate fragment f to node n

    if [TRM-Save(f,m) + RLB-Gain(c+1,f) < 0]
    then De-allocate fragment f from
    node m
    else c := c+1

  elseif [TRM-Save(f,n) + RLB-Gain(c+1,f) > 0]
  then Allocate fragment f to node n
    c := c+1
```

```
end DeletePair
```

It has been demonstrated that each of these algorithms can be implemented so that a call to either requires only  $O(\log m)$  time to construct a new allocation scheme, where  $m$  is the number of nodes in the computer network.<sup>6</sup>

## 5. EXPERIMENTAL RESULTS

To illustrate the applicability of the model and algorithm Alloc, we compared three system designs with respect to a sample set of operations.

The system designs considered are as follows.

Fully Centralized: One central processor with only memoryless terminals at each of 10 MBAs.

Distributed, Fully Replicated Data Allocation: One central processor and a processor at each of the 10 MBAs. The entire database is replicated at each of the 11 nodes.

Distributed, Optimal Data Allocation: One central processor and a processor at each of the 10 MBAs. The data allocation scheme is selected by algorithm Alloc.

The sample application is modeled by a set of 106 different operations, partitioned into six classes:

Class I: On-line device transaction entry and validation, 10 per time unit.

Class II: Manual transaction entry and validation, 10 per time unit.

Class III: Transport of an item from one MBA to another, 10 per time unit.

Class IV: Inquiry on Book Inventory item, 5 per time unit.

Class V: Inquiry on Transaction History, 5 per time unit.

Class VI: Global analysis of Book Inventory and Transaction History, 2 per time unit.

### Results

Following is the cost summary of the three designs considered.

<u>Centralized System (only possible allocation)</u>	
Weighted Cost:	73,000,000
Retrieve Bytes per time unit:	260,000
Update Bytes per time unit:	240,000
Reliability Measure:	0
Prob. of performance (avg.):	.95
Prob. of performance (worst.):	.95
<u>Distributed System, Fully Replicated Database</u>	
Weighted Cost:	600,000,000
Retrieve Bytes per time unit:	0
Update Bytes per time unit:	2,400,000
Reliability Measure:	13,800
Prob. of performance (avg.):	1.0
Prob. of performance (min.):	1.0
<u>Distributed System, Optimal Allocation</u>	
Weighted Cost:	29,000,000
Retrieve Bytes per time unit:	150,000
Update Bytes per time unit:	80,000
Reliability Measure:	4,000
Prob. of performance (avg.):	.96
Prob. of performance (worst.):	.95

The weighted cost and reliability measure are computed using the cost formulae presented in Section 4.1. Also displayed are the expected number of bytes transmitted (for retrieval and update) across the network per time unit and estimates of the average and minimum probabilities of being able to perform an operation at any given moment in time (these values are computed assuming a processor failure probability of .05).

From these results, it is apparent that the optimal allocation scheme for the distributed system is significantly better than either of the two other designs. It is difficult to translate the number of bytes transmitted per time unit directly into estimates of response time. The bandwidth of a local area network such as DEC-NET is highly dependent on the quantity and characteristics of system traffic. Under light-to-moderate saturation, a typical estimate of bandwidth is 250,000 bytes per second. As the system becomes more saturated, the bandwidth deteriorates rapidly.

Consequently, for applications in which the processing intensity reflects time units of one second or less, only the optimal distributed design would seem to provide acceptable performance. Note that this conclusion is valid even if this intensity of processing is achieved only at peak times of the day. On the other hand, for applications in which the appropriate time unit is several minutes or hours, any of the three designs would seem to provide acceptable performance.

Current research on the Level-II evaluators will allow more precise predications of system response time, taking into account network load, routing algorithms, and parallel processing.

## 6. FUTURE WORK

Our research is continuing on two fronts. We plan to use algorithm Alloc to help design actual MC&A systems, and we also shall continue to address the Level-II optimization problem.

Our current approach to the Level-II optimization problem is to develop a manageable space of global strategies for processing the operation templates against fixed networks and data allocation schemes. Once we agree on the strategy space, we will correlate each strategy with expected response times. Preliminary results indicate that for the typical MC&A application, processing delay caused by queuing backlog is relatively insignificant. This result is based on

the assumption of a local area network with high-speed communication links and the assumption that any CPU-intensive operation (e.g., sophisticated analyses) can be given low priority. Therefore, we should be able to calculate response times in a fairly straightforward fashion. Expected response times will then be used in conjunction with the operation templates' evaluation vectors to obtain an expectation of the overall system performance provided by the candidate network under study. Finally, greedy and branch-and-bound type algorithms will be used to generate candidate network topologies for evaluation.

## REFERENCES

1. J. T. Markin, R. B. Strittmatter, H. S. Vaccaro, and W. J. Whitty, "Concepts for Advanced MC&A Systems," Nucl. Mater. Manage. XV(2), 48-54 (January 1987).
2. J. T. Markin, C. A. Coulter, R. G. Gutmacher, C. C. Thomas, Jr., and W. J. Whitty, "Design and Evaluation Methods for an Integrated Safeguards System," Los Alamos National Laboratory report LA-9964, Vol. I (March 1984).
3. W. J. Whitty, R. B. Strittmatter, W. Ford, R. M. Tisinger, and T. H. Meyer, "Prescriptive Concepts for Advanced Nuclear Materials Control and Accountability Systems," Los Alamos National Laboratory report LA-10961-MS (June 1987).
4. T. H. Meyer and P. Helman, "Heuristics for Designing Database Records to Minimize Retrieval Times," Technical Report, Department of Computer Science, University of New Mexico (1987).
5. S. Ceri and G. Pelagatti, "Distributed Databases: Principles and Systems" (McGraw-Hill Book Company, New York, 1984).
6. P. Helman, "A Fast, Optimal Data Allocation Algorithm with Sensitivity Analysis," Technical Report, Department of Computer Science, University of New Mexico (1987); submitted for publication in the IEEE Trans. Software Eng.