# Introduction

## THE PROBLEM

Big science implies that there are only a few large, expensive experiments, and that these experiments are collaborative efforts of many laboratories, universities, and industrial partners. Large accelerators, the space program, and even the financial industry are all examples of this trend. As an example, the U.S. and worldwide magnetic confinement fusion programs have reached the "big science" stage with the advent of the next large generation of machines, TPX (Tokamak Physics Experiment) and ITER (International Thermonuclear Experimental Reactor). Indeed, existing experiments such as DIII-D, TFTR (Tokamak Fusion Test Reactor), JET (Joint European Torus), and Tore Supra are already benefiting from these cross-fertilizing trends.

AS A COROLLARY to this trend, scientists will increasingly interact with the major experiments from their home institutions because travel is expensive and moving everyone to a common site is very disruptive and unpopular. Since the fusion device itself will perhaps be radioactive, interaction with it must be done remotely; this may be done from the near-by control room or from across the world with almost equal ease. However, in these days of terrorists, hackers, and clumsy users, the security of any remote access, and especially of any remote control is of primary importance.

THE DATA from these major facilities must be complete and available to all of the researchers in the field via a variety of user-friendly and automated methods. The proprietary nature of some data must be dealt with to the satisfaction of all parties. Provision must be made for the retraction of data, and the subsequent notification of those who have used the data.

FINALLY, new avenues of data integration and use must be explored to solve the information overload problem in a secure environment.

IT IS THE AIM of this proposal is to show that integration of secure compartmented mode (CMW) and multilevel secure (MLS) workstations, together with commercially-available secure relational databases can solve the above problems, namely

- » *security*
- » *proprietary data*
- » *data retraction*
- » *data overload*

The resulting product must also be user friendly and enhance productivity so that scientists will feel that it is worth the learning curve. The solution to these problems can provide a paradigm for future large science experiments. As a test vehicle for this work, we intend to create a secure international database containing pellet ablation data from many of the world's magnetic confinement fusion devices.

THE DATABASE INDUSTRY BELIEVES that the application of secure technology to non-classified projects is an important new business opportunity and is providing software and technical support to this proposal.

## SECURE UNIX OPERATING SYSTEMS

The trend is for scientific computing to be carried out on UNIX workstations. However, the conventional UNIX operating system is vulnerable to attacks primarily because there is a "superuser" (with ID 0) who can do anything on the system. Many standard programs (e.g., *sendmail*, *telnet*, *ftp*, . . .) temporarily assume superuser privileges in order to accomplish certain tasks such as changing the ownership of files. There are security holes in almost every UNIX network service. In the book *Firewalls and Internet Security — Repelling the Wily Hacker* (William R. Cheswick and Steven M. Bellovin, Addison Wesley, 1994), there is a list of 42 known security "bombs" that can be exploited by hackers. Many more remain to be discovered. Hackers exploit these vulnerabilities to obtain superuser privileges; then they can do anything to the system including the destruction or modification of valuable data. And, it is estimated that there will be 55 million nodes in Internet by the turn of the century, each node a possible attack origin.

SECURE UNIX SYSTEMS come in many flavors, but are all evaluated as being at a trust level of *B1* or higher according to the *Department of Defense Trusted Computer Security Evaluation Criteria* (DOD 52000.28-STD, available from the National Computer Security Center). We plan to use user-friendly MLS and CMW systems that employ an X-Window graphical user interface (GUI). Although these systems look and act like normal UNIX systems, the kernel has been replaced by a trusted set of routines. Furthermore, there is no superuser. All privileges usually assigned to the superuser are split up into about 100 fine-grained privileges. For example, different privileges are needed to override the usual *owner*, *group*, *world* discretionary file access (*DAC*) privilege. Furthermore, according to the *principle of least privilege*, these privileges are only granted for the time that they are required within a trusted program. In addition, many system security operations (e.g., adding a user) require that at least two different privileged users participate in the process. On the Sun CMW system, the usual

root role is split into four separate roles — *administrator*, *information security officer* (*ISSO*), *operator*, and *root*. These roles are severely limited in the tasks they can perform.

THE SYSTEM SECURITY is enhanced by the addition of *mandatory access controls* (*MAC*). In a secure system, the MAC controls provide a matrix of access privileges. The hierarchical *security levels* label each file with a classification (e.g., the military labels Secret, Unclassified, Confidential) that measures the information's value to national security. Each level is also split into *compartments* which correspond to "need to know" categories (e.g., NATO, USAF, project_alpha). Every file in a secure system is labeled with its security level and compartment categories (together called the *security label*), and with its DAC file access privileges that specify read, write, and execute privileges for the owner, group, and world. In addition, many systems also support *file access lists* that list explicit users who may or may not access the file.

ALTHOUGH IT IS COMPLICATED to administer secure UNIX systems, we believe that they offer a much more secure basis than conventional systems, and should be the foundation of any future collaborative computing across networks.

## SECURITY IS A USEFUL TOOL

Security is normally regarded as a necessary evil by those who must employ it for classified work. However, the use of these secure UNIX systems would replace the shaky foundations of normal UNIX systems with a provably (evaluated) secure system that will thwart hacking. We believe that it is necessary to start from a secure system and adjust the parameters to make it more user-friendly rather than to add security to a non-secure platform. As a benefit, the classification levels and compartments can be employed to solve the proprietary data issues, and the audit trail can provide an important integrity function. Other groups are studying shared databases for large projects, but none of them is basing their system on a provably secure platform.

■ *Proprietary data issues*

Experimental data are often fragile and ephemeral things. Usually the data must be processed by its owner to be converted into physical units and must be allowed to age before it is deemed fit for distribution to the rest of the world. Raw data serves very little use to anyone but the person who gathered it.

ONCE DATA is taken, there exists a certain period of time when the data's owner and group should be allowed exclusive use of the data to be the first to publish papers util-

izing the new data. Patents and copyrights serve the same purpose in commercial ventures. But rather soon, perhaps about three months after the data has been reduced from its raw state, it should be available more widely in the scientific community.

OF COURSE, data could be kept secret by its owner by merely storing it on his PC, Mac, or workstation. But, if the system envisioned in this proposal were in place, it would be to the data owner's great advantage to get his data into the main database as soon as possible. Since self-interest is a primal force of nature, a key drawback to the common data pool concept would be removed.

THE ADVANTAGE gained by entering data into the database as soon as possible would be the ability to utilize the rest of the data during the analysis process. The same tools could be used to extract, analyze, and plot new and old data. However, the owner's new data would be marked (classified) *new*, and placed in a compartment assigned to its owner. Then only the owner would have access to the data, and he would be guaranteed (by the security features of the system) that it was secure.

SO, HOW DOES THE DATA BECOME AVAILABLE TO ANYONE ELSE?
After a short period of time has passed (say, one month), the data would automatically be downgraded in level to *proprietary* and its compartment changed to the owner's group so that his colleagues would be able to use it in their publications. It might be that the data is still not processed properly for use by others, so the owner could stop the downgrade-process by responding to a warning one week before the event. The cancellation process should be onerous enough to encourage prompt data reduction. After all, the data obtained from a national or international facility should be available as fast as possible to allow theoretical analysis and to provide proper programmatic direction. After an additional period (e.g., two months), the data label would be downgraded again to *available*, and the compartments would be removed.

BECAUSE THIS DATA STORAGE IS TAKING PLACE ON A SECURE SYSTEM, the built-in audit trail keeps a record of who has retrieved what data. Although the audit trail adds overhead to the system, it can serve an extremely useful purpose. Suppose, for example, that the experimenter who took the data later finds that his instrument was miscalibrated, and that the data were incorrect. By examining the audit trail, everyone who used the faulty data could be notified so that appropriate measures could be taken by those users.

■ *Security for security's sake*

There are real and important uses for security. Most security threats come from legitimate users making a mistake and the users must be protected from themselves. But some forms of attack could be quite subtle. For example, a database query could be supplied with phony data ("spoofed") and incorrect conclusions could result in catastrophic operational modes in future experiments. Accordingly, the security and integrity of the database is vital. Furthermore, the user of the database must be assured that the data is really coming from the database, and that the data is unchanged during the transmission process. Smart-card logins, data encryption and verification, and privacy-enhanced mail are all tools that can be employed to accomplish these ends. The internal integrity and security of the database is assured by the features built into the multilevel-secure (MLS) database engine.

A MORE DIRECT THREAT is posed if the facility or its diagnostics are operated remotely across a computer network. Although this proposal will not address these problems directly, such remote control should also be performed using secure operating systems. The experience using secure operating systems for nonclassified purposes gained in this database task will be directly applicable to the remote control issues.

## THE DATABASE

Currently, most experimental sites have their own proprietary data format — a situation strongly reminiscent of the Tower of Babel. As a result, analysis using data from several experiments requires that a translation protocol layer be inserted between the database and the inquiry tool. Furthermore, most of the databases are not set up to be queried using SQL (structured query language), an ANSI standard. It is essential that the data be stored in MLS relational databases so that standard SQL queries can be used. Provision should be made for the inclusion of compressed pictures and movies in the database.

■ *SQL is a must*

A major advantage of requiring SQL is that a myriad of front-end tools running on all computer platforms can be used to access and analyze the data. The use of commercial off-the-shelf (COTS) software should be strongly encouraged to avoid wasting resources. And the ability to embed SQL calls into Fortran or C programs provides a powerful path towards creating interchangeable custom tools.

■  *Multilevel security*

Multilevel secure distributed databases now represent a mature technology. A secure database engine hosted on a CMW or MLS machine at each major site can serve as an individual node on the data network. Then any query will transparently gather data from each node on the network and return it to the originator. Two-stage commit processes ensure that the database integrity is maintained across the network. By breaking the database across various sites, local customization is possible, and remote data shadowing is possible for added security.

ALL DATABASES have security features in the sense that a given user can only access certain tables and views, and he may or may not be allowed to change the data or to enter new data. However, secure databases go one step further — they allow each row in each table to have its own classification and compartment tag. As a result, the same SQL query will yield different results according to the security privileges of the user. Conventional databases would require that a different view of the data be created, or that a different SQL query be used for each user. More details concerning the differences between single level and MLS-secure databases are discussed in Reference 1.

MECHANISMS will have to be devised that allow the creation and deletion of tables in a secure, yet user-friendly manner.

■  *Metadatabase*

To fulfill future needs, the narrow concept of an experimental fusion database requires extension. By designing a metadatabase in conjunction with the experimental database, the utility and accessibility of the data can be expanded [2]. A metadatabase is primarily a locator tool that provides information at the individual data element level and at higher aggregations of detail, (e.g., technical reports, computer models, etc.) which adds considerable contextual information. The metadatabase would contain a classification scheme and a thesaurus of descriptors and their interrelationships. Additional fields and pointers would be established for subjective information such as the purpose of the experiment and degrees of confidence in the data itself.

MULTILEVEL SECURE TECHNOLOGY contains a configurable audit trail. When used in conjunction with the functionality of a relational database management system (RDBMS), it is possible to keep track of the use of each row of data. From this, use statistics and a mechanism for tracing data elements and the transformations that they undergo can be developed and maintained [3]. This could link each specific data element to the exact row and column position in a specific table in a specific version of a

report thus providing the mechanism for understanding exactly how data were derived, collected, processed, etc.

WITH THE METADATABASE AS AN INTERMEDIARY, views would be set up to partition the data in different dimensions (e.g., by machine, by diagnostic, by physical effect, etc.) or linked to similar atomic physics and material databases. The browsing, searching and selection of data elements would occur in one location, but would allow selection and use of information from other fields in the fusion analysis.

## DATA OVERLOAD

Using a large, constantly changing database for research purposes is a challenging and time-consuming process. Data overload is an ever-present worry. Without assistance, each scientist might have to spend days each week examining the database to see what is new and what is of interest. Technology developed at Oak Ridge for the Federal Aviation Administration (FAA) can solve this problem.

ON EACH SECURE DATABASE HOST, a Server would take SQL requests from Clients across the network. The SQL represents the Client's interest or information requirement. A Client may have several active requests at any time. Whenever any data were added to the database (or updated) that matched the query, the data would be sent to the Client. The Client need not be a human sitting at a terminal. For example, a daemon Client would accept data for scaling law fits, automatically add it to the local database, refit the augmented data, and inform the scientist via E-mail that a new result was available!

THE SERVER is smart enough to poll its Clients occasionally to be sure that the connections are still available, and to attempt to re-establish the logical connection if it is not. The burden for information retrieval now lies with the server. Clients are informed of changes when they are up to receive them.

# Objectives

The work on this project divides itself into several natural parts:

- *the database*

- *the security aspects*

- *the networking*

- *the user-interface and analysis tools*

- *the client-server technology*

## THE DATABASE

This proposal will utilize data from pellet injection experiments in the US, France, England, Russia, Germany, and Japan. A key problem that must be tackled head-on in this proposal is that the database must be attractive to its users in spite of the security apparatus. Motivation of users is therefore a primary goal.

DATA in the database must be complete in the sense that all diagnostics for a given shot should be included. The usefulness of the data increases rapidly if more shots and more fusion devices are included. In addition, views of the fundamental data tables should be provided so that the data can be presented in ways that make queries easier. For example, data may be organized by machine, by diagnostic, or by physical effect. We hope that users will be able to use the mechanisms of our data system to urge their colleagues to get raw data analyzed so that more complete data sets are available.

IF THE DATABASE MANAGEMENT SYSTEM (DBMS) is going to be used for the reduction of raw data, individual users must have the ability to create, delete, and populate temporary tables. However, raw data is not useful to other than its creator, so these tables should have very limited lifetimes. A mechanism must be created to integrate new types of data into the database in a manner that enables it to be queried and to be placed into appropriate views. This act would trigger the client-server mechanism to rerun its queries on the newly-added data.

■ *Database engines*

WE HAVE SEVERAL SECURE DATABASES that are designed for our MLS and CMW systems. Initially, we plan to use *Informix*®on the HP MLS 9.09 platform because we know how to allow access to this system from non-secure hosts at any user-selected security level. This issue will be expanded upon in a later section.

## SECURITY ASPECTS

The U.S. Government has made a major investment in secure computing and networking. Although security is still important in the military venue, it is appropriate to make use of these technologies in other areas. The fusion database is the perfect example of a problem that must be solved and that fits well into the multi-level security paradigm. The challenge is to set the security safeguards so that security considerations do not become burdensome. We feel that starting with a fully-secure system and making intelligent choices to ease some of the constraints is more cost effective and more secure that the alternative, namely, beefing-up security on a non-secure system.

IN GENERAL, users will access the database across an untrusted network from untrusted hosts and will be able to log into the secure host at a single level of security. Several issues arise:

- *Login IDs and passwords must be secure.*

- *Data must be protected from unauthorized access and certainly from unauthorized changes.*

- *The authenticity of received data must be assured.*

- *The secure host will be subject to attacks from the network.*

- *Audit trails will assure security and allow data clients to be notified of data set retraction and/or changes.*

WE EXPECT to institute a form of protected logon, either via Kerberos (an MIT software product) or SmartCards, a credit card-sized device that generates a new, secure password every minute. Data encryption using Kerberos, privacy-enhance mail (PEM), or other encryption techniques may be optional, but if encryption is not used, a security wrapper containing a special checksum of the data in each packet will have to be used.

TO PROTECT the secure system from network attack, we will employ a firewall — a computer that separates the secure network from the rest of the world. Users log in to the firewall, and then into the database computer.

THE ISSUE of how to connect a user on a non-secure host with a secure database running on a trusted host is a difficult one. On the HP MLS 9.09 system, there is no problem — the system asks the user what security level is desired during the login process. But on our Sun and DEC CMW systems, this option does not exist. Therefore, we will have to do research to determine a method of achieving connections at more than one

level. For example, we might be able to configure the firewall machine (an AT&T 3B2 with a B1-secure operating system) as a multilevel host, and do the level switching on the firewall.

HOWEVER, in the case of the Sun CMW system, the operating system has been designed so that multiple workstations running CMW 1.1 can be configured so that they act as if they were one distributed system. All of the user accounts and security attributes are administered on one workstation (the NIS master) and promulgated across the network to the other workstations (clients). Using this configuration, complete multilevel-secure communication is maintained among the machines. We have two CMW Sun workstations configured in such a master-client configuration.

THE DISTRIBUTED secure CMW environment is designed to use a secure network. However, if we encrypt all communications among these machines, we believe that such a configuration could be extended in a secure manner between remote sites located on the Internet. We propose to try this configuration as another approach to solving the multilevel system access problem.

## NETWORKING

The secure workstations are all connected by a separate fiber-optic network located in the Data Systems Research Division of Martin Marietta Energy Systems (MMES). The availability of a separate network ensures that the heavy network traffic that results from a distributed database will not impact the rest of DSRD or the MMES computer network. Also, it allows us to experiment with various security options in private. The secure network can be disconnected from the rest of the world, or connected through a firewall computer to the rest of Internet via the MMES network.

NETWORK BANDWIDTH is precious, especially over Internet, so various data compression algorithms will be tried. Profile data and plots can require large bandwidths if left uncompressed. We have tried accessing our secure system from France, and have learned that if possible, any X-based front end should run on the user's machine to avoid sending verbose X commands across the network. The situation may improve when compressed X protocols become more readily available.

## USER-INTERFACE AND ANALYSIS TOOLS

We plan to use *Wingz*®a spreadsheet from Informix  as a front-end interface tool to the database. Spreadsheets are easily used and are familiar to most members of the scientific community. In addition, we will investigate the use of 4GL tools such as *Uni-*

*face* or *Power Builder* to provide SQL query generators, user-defined views, and data entry forms that run on any platform. Eventually, we hope to be able to use *Mosaic* as a front-end tool for routine queries.

USER FEEDBACK is vital since users must be convinced that it is cost-effective and productive to use our database system. Some travel to user sites to gather user feedback and to proselytize is included in the budget.

## DATA OVERLOAD

The FAA-developed client-server technology that will be used to solve the data overload problem interrelates three entities:

- *The DBMS notifies the server upon changes to the database (e.g., insertions and updates).*

- *The Server receives registration of interest in types of information, security levels, and information compartments from Clients.*
  *Upon notification by the* DBMS, *the server sends updated information to authorized clients who have registered an interest in the data.*

- *The Client registers data interest and security/authorization level with the Server. The Client also receives from the Server updated data of interest.*

THERE ARE SEVERAL ADVANTAGES to this approach. It obviates the need for the user to query the DBMS to determine what data have changed and what data are of interest. The user may set up a daemon process that will update itself when new data arrives and analyze it without the user's intervention. The client-server approach reduces DBMS processing and data transfer load by delivering only wanted data to interested users. In addition, the Server can fulfill low-priority queries at night when user load is light.

THE SERVER is smart enough to poll its Clients periodically to see if they are still logically connected; if not, it will re-establish contact.

# Technical Approach and Plan

Although it sounds straight-forward to say that we will install a database on a computer, installing anything on a multilevel secure operating system is a difficult process that may encounter numerous obstacles. Installing COTS programs that are not designed for these secure systems is even harder. For example, error messages are purposely obscure to prevent classified information from being revealed via a covert channel. Accessing these systems from remote non-secure terminals, in a secure, but user-friendly manner, is an unsolved area of research. Fortunately, we have developed contacts at the major hardware and software vendors that we believe will help us to overcome these problems. The DSRD secure database team has been working with these systems for over a year on most of the available platforms, and we are now well versed in the field.

Therefore, barring unexpected surprises, we envision the following work breakdown structure for the first year of this project:

■ *The secure platform:*

- *Install the database on MLS and CMW secure and non-secure platforms on the DSRD security network*
  - » *Enable network security for remote access*
  - » *Wrappers*
  - » *Smart Cards*
  - » *Set up the classification levels and compartments*
  - » *Adjust the security of the system so that it is user-friendly but still secure*

- *Determine methods of connecting to secure hosts from Internet at more than one security level.*

- *Investigate using the audit trail to notify users of data retraction.*

- *Investigate encryption techniques for secure transfer of data across public networks.*

■ *The database:*

- *Design the pellet ablation database, set up the tables.*

- *Use Wingz and Query generators as front-end tools.*

- *Administer the database.*

- *Collect pellet ablation data from international sites.*

- *Preprocess data so that it is suitable for import into the database.*

■   *The metadata:*

- *Incorporate metadata into the usual experimental facts and figures.*

■   *The Client-Server technology:*

- *Set up remote, unattended access to the database over Internet.*

- *Create a secure SQL Server to handle remote requests and an unsecure Client to generate requests and to receive the answers.*

- *Integrate them with the DBMS.*

■   *User involvement:*

- *Canvass user community to obtain desires.*

- *Set up user tutorials.*

- *Run a pilot program at perhaps one user site.*

# Implementation Plan

### First Quarter

- *Install Informix Online Secure and Hypertools (containing Wingz) on the HP MLS platform. Hypertools is an ordinary COTS program, it may be difficult to install it so that it is accessible by all users at all security levels. This is the measure of success.*

- *Install Informix Online Secure database on one Sun CMW system.*

- *Design the pellet ablation database, create it in Informix, and populate it with enough data to make it useful to scientists in the field. The data will be entered using Wingz and transferred to Online Secure via Datalink.*

- *Create a front end tool for entering data into the database.*
  - » *Power Builder and Wingz scripts.*

### Second Quarter

- *Install encryption between the two distributed Sun CMW machines.*

- *Replicate the pellet ablation database on this system.*

- *Test access to the database across the network. Are the Max Six encrypted packets compatible with Internet?*

- *Test the security of this system by moving half of it outside of the firewall and trying to penetrate it.*

- *Evaluate the auditing capabilities of the various secure databases to test the feasibility of the data retraction feature.*
  - » *Add metadata to the databases.*

- *Test the user interface using scientists who use these data. This test is not intended to be exhaustive, only a way of assessing where problems occur, and what improvements are desired.*
  - » *Can they easily enter data?*
  - » *Can they easily extract data?*

- *Determine whether they think that the system is worth the learning curve. How can it be improved?*

<u>Third Quarter</u>

- *Add the Client-Server unattended query component to the systems.*

- *Visit several user sites to demonstrate the system and to train interested users.*

- *Set up international user access to the system.*

- *Have scientists test remote access to the databases on both the MLS and CMW systems.*
  - » *Evaluate front end tools*
  - » *Test the desirability of running the front end tool on the user's machine or on the database machine.*

- *Turn on the security audit trail and develop audit reduction strategies.*

<u>Fourth Quarter</u>

- *Evaluate user interaction with the system.*
  - » *Are the data interesting enough to use?*
  - » *Are proprietary interests protected?*
  - » *Is the security overhead tolerable?*
  - » *Does using the system improve the user's productivity?*
  - » *Are the front-end tools sufficiently flexible and attractive?*

- *Write up final report.*

Some discussion of the measures of success is in order. Fulfilling the above milestones is a significant and difficult challenge. While secure systems and secure databases are commercially available, actually getting them to work is non-trivial.

There exist no front-end database tools designed to operate on these CMW and MLS secure systems. Utilizing COTS software products on these systems without violating the security of the system (by giving the COTS program too many privileges) is a challenge. We have about 20 pages of tips on how to get WordPerfect to run successfully on the CMW Sun, and have only been partially successful in the effort. In general, support from the manufacturer of both the hardware and software is essential if success is to be achieved. We now have the requisite contacts to obtain the information that will be needed to achieve success.

It is difficult to evaluate the overall success of the project. Working on a system without the strong security that we will provide will surely be easier and more "user friendly." However, we believe that the requirement for a truly secure system will become more imperative as time goes on. So, it would be unfair to compare the usability of our system to similar non-secure systems; a more absolute standard is required. At the end of the first year, we expect to have all of the essential ingredients of our system in place for a few brave scientists to try. We need these users to generate feedback so that we can flesh out the components in an attractive manner.

The ultimate measure of success is repeated use by scientists. This implies that the learning curve is worth the effort and the user's productivity is improved.

In order to reach this goal, the data in the database must be interesting. And the database must be dynamic in order to lure users and in order to test the Client-Server access schemes. Other database efforts have failed because the data were fully-analyzed, "famous" shots. In other words, those people who were interested in the data already had seen it before it was entered into the database. The database was characterized as "boring," and no one used it.

We are keenly aware of the above problem which is why we have included money for proselytizing. Nonetheless, we do not think the database will reach a critical mass during the first year of the program, and the front-end tools will still require more development. So, we expect that the second year will be devoted to enhancing the attractiveness of the data and of the database system. The first year will provide a proof of principle.

<u>Goals for Year Two</u>

- *Have new, unanalyzed data entered into the database to test the "declassification" procedure and to fully implement protection of proprietary data.*

- *Improve the front-end interfaces, especially those running on the user's computer.*

- *Fully implement the audit trail including the capability for data retraction..*
  - » *Triggers may have to be used to keep track of access on a row by row basis.*
  - » *Continuous audit trail reduction must occur in order that the audit trail not grow larger than the database itself.*

- *Implement Mosaic as a front end to the CMW system for more casual queries.*

- *Test the distributed database concept by splitting the database over several secure host machines.*

- *Determine the success of this effort as judged by the users.*

# Data Requirements

The data that will be used will be obtained from pellet-injection experiments at major labs all over the world. Some of the devices that will be included in the pellet ablation database are Heliotron-E (Kyoto), DIII-D (La Jolla), T-10 (Moscow), ATF (Oak Ridge), JET (Culham), W7-AS (Garching), TFTR (Princeton).

THE FORM AND CONTENTS of the database will be determined by consultation with the members of the various pellet injection experiments. The data will be collected in collaboration with the Fusion Energy Division of Oak Ridge National Laboratory.

# Supporting Facilities

## A PARTNERSHIP WITH INDUSTRY

Multilevel-secure database systems represent a mature technology, but are still complicated, especially when embedded in a distributed CMW environment. Therefore help from industry is essential. One of the major database vendors, Informix, has expressed a desire to participate in this project with the hope that the paradigm developed by this project will have commercial value in other venues as well as in different "big science" projects.

ACCORDINGLY, Informix has agreed to provide several copies of their Online Secure database engine, together with technical support, for the duration of this project. Informix has the capability of storing "blobs" which are large chunks of any sort of data. It also can store picture image files (up to 2 Gb in size) either directly in the database or as pointers to some other storage location. Several 4GL tools will be provided to allow us to create menus, views, and other user-friendly interfaces to the database. In addition, the Informix database is tightly coupled to the *Wingz* spreadsheet which has excellent graphics capabilities. *Wingz* will also be supplied by Informix, and will be used as another primary user interface to the data. The commercial value of these contributions is over $100,000.

## OAK RIDGE RESOURCES

The Data Systems Research Division of Martin Marietta Energy Systems is contributing the use of its secure workstation network. This network currently has secure Sun (CMW), AT&T (MLS), DEC Alpha, and Hewlett-Packard (MLS) operating systems. CMW for the IBM Power PC will be obtained when it is available. In addition, there is

a non-secure Sun workstation and two non-secure NeXT workstations together with many PCs. Since these systems are used for security-testing purposes, there is no actual classified data on them, and the workload is very light. The network is implemented as a separate fiber-optic network connected to the rest of the Energy Systems network through a "firewall" computer. The fact that the secure computers are on a separate network will allow us to test various security features without impacting the rest of the MMES computer network.

MARTIN MARIETTA ENERGY SYSTEMS has broad experience in the areas of distributed computing and networking, as well as the administration of large databases.

# Management Plan

The number of people working on this project is fairly small, and they are all concentrated in one location, so a complex management scheme is not required. The technical work will be led and supervised by the Principal Investigator, James A. Rome. The financial aspects will be administered by Patricia W. Payne.

# References

[1]. Patricia W. Payne, *Issues in migrating from single-level to multi-level (MLS) databases*, 16th Department of Energy Computer Security Group Training Conference, Denver Colorado (May 3–5, 1994).

[2]. *Database systems: Achievements and Opportunities*, the "Lagunita" Report of the NSF Invitational Workshop on the Future of Database Systems Research, Palo Alto, CA (Feb. 22–23, 1990).

[3]. J.M. Griffiths and K. Kertis, *Sharing information via metadatabases*, Proceedings of the Tactical Technologies and Wide-Area Surveillance International Symposium, Chicago, IL (November 2–5, 1993).

# Biographical Sketches

**James A. Rome**, Senior Scientist, Fusion Energy Division, Oak Ridge National Laboratory

*Principal Investigator*

S.B., Electrical Engineering, Massachusetts Institute of Technology, 1964
S.M., Electrical Engineering, Massachusetts Institute of Technology, 1967
Sc.D., Electrical Engineering, Massachusetts Institute of Technology, 1971

*Martin Marietta Energy Systems, Inc., in-house courses*:
  Project Management; Negotiating Skills; Program Development.

With a strong background in theoretical and experimental research, Dr. Rome uses an interdisciplinary approach to projects. Current research includes data analysis of air traffic flow patterns for the FAA, and the study of secure distributed databases running on CMW and MLS secure workstation platforms. In the fusion area, Dr. Rome developed most of the theory for neutral beam injection into toroidal plasmas including deposition, thermalization and loss regions. He is an expert at following charged particles in complicated magnetic geometries and designing magnetic configurations (stellarators) to obtain specific physics results. Dr. Rome originated the computational techniques needed to build, measure, and assemble the complicated helical coils in the ORNL Advanced Toroidal Facility. He is Editor of the bimonthly newsletter *Stellarator News*. He is also President of Scientific Endeavors Corporation, a company that specializes in scientific graphics. Dr. Rome is a Fellow of the American Physical Society.

**Recent Publications:**

J. A. Rome, "Orbit topology in conventional stellarators in the presence of electric fields" Nuclear Fusion, in press, (1994).

James A. Rome, Larry R. Baylor, and Patricia W. Payne, "Using Secure Databases for Unclassified Purposes," 16th Department of Energy Computer Security Group Training Conference, Denver, CO (May 3–5, 1994).

K/DSRD-1584  "Department of Energy Data Management Security Guideline Information" (December, 1993).

J. B. Wilgen, et al., Fluctuation and modulation transport studies in the Advanced Toroidal Facility (ATF) torsatron," Physics of Fluids B, **5** (July 1993) 2513

K/DSRD-1098  "Analysis of the National Airspace Capacity" (September 30, 1992).

K/DSRD-1190  "FAA Data Analysis User's Manual" (September 30, 1992).

**Patricia W. Payne**, Data Systems Research and Development, Martin Marietta Energy Systems, Inc.

B.A., Sociology, University of Tennessee, Knoxville, 1977
M.S., Planning, University of Tennessee, Knoxville, 1988

Ms. Payne specializes in Software Engineering with a focus on information engineering techniques. She employs data modeling and interface design tools to develop user-friendly interfaces for trusted and untrusted relational databases. She is currently working on database migration problems, database configurations and interface designs for trusted relational databases (Informix/Online Secure, Oracle Trusted, and the Sybase Secure Server) on the secure DSRD network. Recent projects include database administration of a 12 Gigabyte database; data modeling, design and development of user interface for the NAMO Naval Aviation Logistics Data Logistics Analysis (NALDA) System; and strategic planning for a document tracking system. She is manager of the DOE database security task.

**Johnny S. Tolliver**, Computing Applications, Oak Ridge National Laboratory

B.A. with highest honors, Physics, University of Tennessee, Knoxville, 1976
M.S., Physics, University of Tennessee, Knoxville, 1980
Ph.D., Plasma Physics, University of Tennessee, Knoxville, 1984

*Martin Marietta Energy Systems, Inc., in-house courses*:
  Parallel Processing, including hands-on experience with Sequent shared memory architecture and the Intel iPSC/2 Hypercube architecture
  Artificial Intelligence (AI), with exposure to LISP and other AI programming languages and expert systems
  Practical Solution of Differential Equations, stressing analytic solution methods

Dr. Tolliver is the R&D Group Leader in the Computational Physics Section of the Computing Applications Division of Oak Ridge National Laboratory. By training he is a computational plasma physicist. He is currently involved in modeling power absorption in high-density inductively-coupled plasma semiconductor processing reactors. In recent years he has broadened his areas of expertise. He was Lead Analyst of a 5-member team developing UNIX/X/Motif-based software written in C++ to implement a rule-based artificial intelligence system supporting coordinate measuring machine (CMM) metrology for CMM inspection of complex part shapes. For the past year he has also performed trusted operating system and trusted database research and training

for Department of Energy trusted database research tasks and Automatic Information Systems Security training efforts.

Dr. Tolliver has installed Privacy Enhanced Mail on several computer systems, and is a skilled UNIX system administrator.

**Ron W. Lee**, Engineering Physics and Mathematics, Oak Ridge National Laboratory

B.S., Information and Computer Science, Georgia Institute of Technology, Atlanta, Georgia
M.S., Computer Systems, Air Force Institute of Technology, Wright-Patterson AFB,Ohio

Mr. Lee is a Computing Specialist who has led the development of several object-oriented systems, including near real-time message matching and network routing facilities (API and utilities) for the FAA, a U.S. Army planning tool for graphic display of troop and cargo movement data, a geographic data presentation server and API for the U.S. Air Force, and a library of C++ classes for general purpose client-server operation. One major focus has been software reusability and quality. He is an expert on UNIX operating systems and networking.

**Larry R. Baylor**, Fusion Energy, Oak Ridge National Laboratory

B.S., Physics and Electrical Engineering, Iowa State University, 1981
M.S., Electrical Engineering, University of Tennessee, Knoxville, 1984
Ph.D., Physics, University of Tennessee, Knoxville, 1989

Dr. Baylor is responsible for performing and analyzing pellet fueling experiments with the many pellet injectors that ORNL has installed on fusion machines throughout the world. He is an active participant in these experiments, both by extended visits to the experimental sites, and by remote interaction over Internet. He is currently collaborating with other scientists around the world to formulate a pellet ablation database for use in designing future pellet injection systems.

# Budget

| 1. Direct Labor | Year 1 | Year 2 | Total |
|---|---|---|---|
| Researcher (hours) | | | |
| Principal investigator (Rome: 435@$76.79, 290@$82.40) | $33,404.00 | $23,896.00 | $57,300.00 |
| Database administrator (Payne: 435@$63.00, 435@$65.50) | 27,405.00 | 28,493.00 | 55,898.00 |
| Network (Tolliver: 435@$76.79) | 33,404.00 | 0.00 | 33,404.00 |
| Experimentalist (Baylor: 145@$76.79, 145@$82.40) | 11,135.00 | 11,948.00 | 23,083.00 |
| Administrative Support (Dailey: 145@$63.00, 145@$65.50) | 9135.00 | 9498.00 | 18,633.00 |
| Client-Server (Lee: 435@$76.79, 218@$82.40) | 33,404.00 | 17,963.00 | 51,367.00 |
| Total Direct Labor | $147,887.00 | $91,798.00 | $239,685.00 |
| **2. Total Hours** | 2030 | 1233 | 3263 |

| 3c. Equipment | Year 1 | Year 2 | Total |
|---|---|---|---|
| Power Builder (Enterprise version) | $6,010.00 | $0.00 | $6,010.00 |
| Total Equipment | $6,010.00 | $0.00 | 6,010.00 |
| **3d. Supplies** | | | |
| Printing/binding | $160.00 | $160.00 | $320.00 |
| Editing | 125.00 | 125.00 | $250.00 |
| Postage | 30.00 | 30.00 | $60.00 |
| **3e. Travel** | | | |
| Quarterly meetings (4 trips, 2 nights) | | | |
| Air fare (Knoxville, Berkeley) | $5,120.00 | $5,120.00 | $10,240.00 |
| Per diem | 456.00 | 456.00 | 912.00 |
| Lodging | 816.00 | 816.00 | 1,632.00 |
| User-training and feedback sessions (3 trips, 4 days) | | | |
| Air fare | $3,660.00 | $5,000.00 | $8,660.00 |
| Per diem | 570.00 | 700.00 | 1,270.00 |
| Lodging | 840.00 | 900.00 | 1,740.00 |
| Total Travel | $11,462.00 | $12,992.00 | $24,454.00 |
| **3f. Other (N/A)** | | | |

## Cost Sharing

| Item | Total (both years) |
|---|---|
| *Informix Contributions* | |
| 2 copies of Informix Online Secure | $42,280.00 |
| 1 Copy of Informix Online | $21,140.00 |
| Informix Development Toolset | $11,380.00 |
| 1 Copy of Wingz | $9,680.00 |
| 1 Copy of Hypertools | $1,000.00 |
| Net/Client Tools | $3,530.00 |
| Vendor beta software | $5,000.00 |
| *Data Systems Research Division Contributions* | |
| Smart Cards and Reader (DSRD) | $1429.00 |
| Secure-ID Server (DSRD) | $2,950.00 |
| The value of the DSRD security network that will serve as the platform for this project is hard to calculate, but is worth at least several hundred thousand dollars | |
| **Total cost sharing** | **$98,389.00** |

## Cost Summary

| Item | Year 1 | Year 2 | Total |
|---|---|---|---|
| 1. Direct labor | $147,887.00 | $91,798.00 | $239,685.00 |
| 2. Total hours | 2030 | 1233 | 3263 |
| 3a Subcontractor | 0.00 | 0.00 | 0.00 |
| 3b Consultant | 0.00 | 0.00 | 0.00 |
| 3c Equipment (software) | 6010.00 | 0.00 | 6,010.00 |
| (shared costs not in item 4) | 98,389.00 | | 98,389.00 |
| 3d Supplies | 315.00 | 315.00 | 630.00 |
| 3e Travel | 11,462.00 | 12,992.00 | 24,454.00 |
| 3f Other | 0.00 | 0.00 | 0.00 |
| Subtotal | $264,063.00 | $105,105.00 | $369,168.00 |
| 4 Indirect costs (41%) | 67,926.00 | 43,093.00 | $151,359.00 |
| 5 Other costs | 0.00 | 0.00 | 0.00 |
| 6. Subtotal | $331,989.00 | $148,198.00 | 480,187.00 |
| 7 <Cost sharing> | -98,389.00 | | -98,389.00 |
| 8. **Total costs** | **$233,600.00** | **$148,198.00** | **$381,798.00** |