# Overview of a TRW Pilot Program

William A. Olsen
TRW
Columbia, MD

## Abstract

Recently TRW fielded a prototype system for a government customer. It provides a wide range of capabilities including data collection, hierarchical storage, automated distribution, data analysis, and product dissemination of imagery information. The system is capable of processing thousands of data sets and storing over 400 Gigabytes (GB) of data per day. Over 100 users are supported in a distributed system covering eleven geographical areas. The development effort spanned two years and involved the integration of over 200 COTS/GOTS products. The data that is collected, stored, and disseminated is composed of large image file data sets (from 700 MB to three GB per data file) and associated metadata. Initial customer requirements of a two-year retention window and access latency of less than ten seconds dictated three-tier storage architecture.

At the NASA/IEEE Mass Storage Conference TRW will discuss the following:

- development of the prototype in terms of software, hardware, and systems architecture;
- lessons learned from development, integration, and operation; and
- TRW's current initiative, using the lessons learned from the prototype effort, for the creation of a laboratory dedicated to furthering developments in high performance data storage, processing, and distribution systems.

System architecture development will be discussed in terms of hardware selection across the categories of high performance UNIX servers and workstations and high performance robotic libraries. The selection of COTS products discussed will include the categories of Storage Management Software, database products, system management products, and desktop/analysis products.

This paper will address software that was developed to facilitate integration including:

- software integrating traditional database functions with COTS storage management products to provide a multi-tier storage capability satisfying customer requirements in terms of performance and cost;
- system management tools used to distribute processing of data sets; and
- software (glueware) integrating numerous desktop analysis and collaboration tools with the data storage and distribution process.

In terms of software and hardware integration the paper will address requirements "creep", hardware performance in the initial operational environment, system/software tuning, the reality of vendor API specifications, and the complexities of developing "middleware" in a very heterogeneous hardware/ software environment.

The operational lessons learned address the life-cycle support and administrative complexities of a distributed storage and processing system. Issues covered will include the following:

- the management of multiple tiers of storage (on-line, near-line, off-line);
- the difficulties of volume management, media content management, and media re-use and the strategies – from original approach to current implementation – used to address these issues; and
- achievement of the needed system flexibility (scalability) to handle significant extensions and changes of requirements while supporting existing operational capabilities.

The vision of TRW in Columbia is to create a cooperative laboratory environment that supports distributed computing, hierarchical storage management, and data distribution. The objective of the laboratory is to study and develop high performance system architectures to distribute, manage, and process large amounts of data for a variety of applications. The laboratory will use the pilot program as a baseline to build an infrastructure and delivery system. This will facilitate an environment for the development and integration of COTS/GOTS components to produce a service based architecture that can support many different mission environments. The first step in achieving these goals involves the foundation of a laboratory available for use by scientists, software developers, vendors, and systems integration engineers. When completed the laboratory will consist of hardware and software supplied by the government, vendors, and TRW. The environment will allow vendors, software developers, and system integrators to demonstrate products and components working in an integrated system. Considerable attention will focus on evaluating

COTS products in the areas of storage management, databases, desktop tools, and user applications.

## Introduction

A primary goal for the pilot program was a proof of concept: could a system be integrated from COTS and GOTS components that would meet the mission requirements of analysts/users and provide an end-to-end softcopy environment. The proof of concept was the rapid development and deployment of a distributed system for the purposes of collecting, storing, analyzing, and distributing imagery data and reports. The effort was an attempt to bring a Web-based desktop and tool set to the analysts. The existing softcopy imagery analysis system was composed of costly, custom hardware and software and access to it was very limited. The integration of numerous COTS software products into a Web-based interface dramatically changed system development, life-cycle support, and user productivity. The user now has timely access to Terabytes (TB) of data and a powerful set of tools to process, analyze, collaborate via high-speed network, and generate and distribute high quality reports.

The topics discussed herein cover all phases of the prototype program. Goals from the perspective of management and system use and maintenance will be described. The discussion of system architecture design and development based on system segmentation and requirements analysis will cover issues including functional capabilities, scalability factors, customer preferences, price/performance issues, and overall component compatibility with desired COTS hardware and software products. Detailed examination of products in the areas of database, storage management software, and software products that provide access to data and metadata in a distributed processing environment were integral to the successful prototype system.

### Management Goals

Management goals focused on rapid capability insertion, use of commercial technology, reduction in the cost of the systems and the life-cycle costs, and architecture that allowed for continuous technology insertion, and increased analyst/user productivity.

**Reduced Development Time.** Management goals included a reduction in system and software development time, to be achieved by integrating existing COTS/GOTS software and hardware products. The original system took over 5 years to produce, the average cost per workstation was over one million dollars, and the maintenance price for all the systems was over twenty million annually. The prototype went into operational testing eighteen months after contract award, and the average cost per workstation was one hundred thousand dollars (including network costs and COTS costs). The analysts/users were so pleased with the performance that the system transitioned from a "proof of concept" to an operational prototype (24 hours a day by 7 days a week) and remains in operations for 100 users as the Government procures the next generation systems.

**Enhanced Production Value and Increased Timeliness.** The system was designed to accelerate production value from new system users. Immediate user effectiveness was expected to result from the friendly interface and enhanced user effectiveness was expected to result from access to improved analytical tools and collaborative, data-sharing tools. Increased production value from experienced users was also a goal. The availability of collaborative tools and easy to use office automation tools allows users to produce quality reports in a timely manner, measured in hours rather than days.

**Reasonable Price/Performance Ratio.** The optimal price/performance ratio was to be achieved by maximizing the use of COTS software and hardware products, providing for decreased life-cycle maintenance costs, and making prudent use of object technology. As mentioned above, the price performance was reduced from $1M per seat to $100K per seat. Metrics are being collected on the maintenance costs, but so far the maintenance of the systems is being accomplished by a team of four engineers and COTS maintenance agreements.

### Operational Goals

Operational goals focused on providing high performance, reconfigurable COTS-based workstations and standard user interfaces. Because of the object-oriented design and the use of JAVA for the glueware, each site was easily configured to meet their mission needs. One site needed more on-line storage to handle higher data requirements. Another site needed additional cache hardware to support their near-real-time requirements. Another site required different high performance monitors to satisfy unique analysis requirements. The system was developed on a SUN platform. We were able to re-host the software on a DEC Alpha in less than a day and on a SGI system in four hours. The prototype proved that systems can be developed that are hardware and specific COTS independent if that is a design objective from the beginning.

**Centralized Management and Flexible Configuration.** Due to the geographical dispersion of the users, central management of software baselines was desirable. The system therefore should be capable of making changes to the users' desktops from a central location. Increased flexibility in desktop configuration was needed to make changes efficiently and effectively on a per-user basis. By monitoring each user's desktop configuration, system administrators can remotely assess the hardware and software status of individual users.

**Scalable Solutions.** The system's design should allow it to scale to hundreds of workstations at multiple sites by making sound architecture decisions and efficient use of WAN bandwidth. This goal was accomplished. The proof of concept was to demonstrate a system for 22 users working at two sites. This architecture supported the growth to 100 users at 11 sites without change to the hardware or COTS products.

**Design and Structure.** A client-server structure is optimal to support the distributed environment. Open systems design supports the client-server structure while allowing for a heterogeneous computing environment. The philosophy and intent was to be able to duplicate the pilot program in another environment independent of the underlying hardware. This would allow for incremental upgrades of various functional areas based on emerging technologies. Because of the open system design, the architecture supported PCs, Macs, SUN SPARCs, DEC Alphas, SGIs, and other hardware all running on the same LAN with users able to communicate and collaborate. Each site had unique legacy applications they used to accomplish their mission. The design allowed for integration of these applications into the standard desktop to meet each site's individual requirements.

## System Requirements

The prototype system was designed to address the need for imagery analysts to examine data and generate reports in an easier to learn and easier to use environment. Specific needs that the prototype system addresses include an intuitive set of Web-based software tools made up of COTS/GOTS products; the ability to share data and enhance collaborative team efforts; a fast system response time; and a flexible configuration of the desktop tools. Some of the other system requirements are described below.

**Functional/Operational Requirements**

The system provides access to and delivery of image data to analysts worldwide, necessitating that data be readily available and accurate.

**Availability.** The prototype needed to be highly available, allowing for the possibility of 7 days per week, 24 hours per day operation and user access to data via redundant paths/links.

**Fault Tolerance.** The system incorporated a high degree of fault tolerance based on the requirement that data entering the prototype system not be lost or unrecoverable when archived into the mass storage subsystem.

**Data Integrity.** A high degree of data integrity was required due to the detailed level of analysis conducted by the user on the visual data. Additionally, the customer required that there be no loss of previously available functional capabilities and that the system's capabilities be enhanced by adding new analysis tools. All data bits for each record were used in complex algorithms to produce data sets for different analysis applications. In the past, Cray computers were required to perform the complex mathematical calculations.

**Performance Requirements.** The system's performance requirements reflect the need for two-year storage of imagery records and a fully integrated record management system.

**Multiple Access Timelines.** The system is required to support multiple tiers of access latency. Users need some data to be displayed virtually instantaneously, while other requests were not as time critical. The three data access requirements were stated as (1) access within ten seconds of a request, (2) access within three minutes of a medium priority request, and (3) access within twenty minutes of a low priority request.

**High Throughput.** The data source for the system is a high bandwidth channel with a burst rate of 155 Mbps and a sustained rate of over 100 Mbps, thereby requiring a high performance on-line and near-line mass storage systems.

**High Capacity Storage.** The system required high capacity, long-term storage. Hundreds of gigabytes of data are stored daily, with a two-year retention period for the data. The system processes thousands of requests per day from a geographically dispersed set of users, requiring a high performance, distributed database to direct requests and manage the distribution of data to the appropriate user.

**Architecture (Hardware)**

The following components represent the foundation for a high performance, enterprise-wide information management infrastructure.

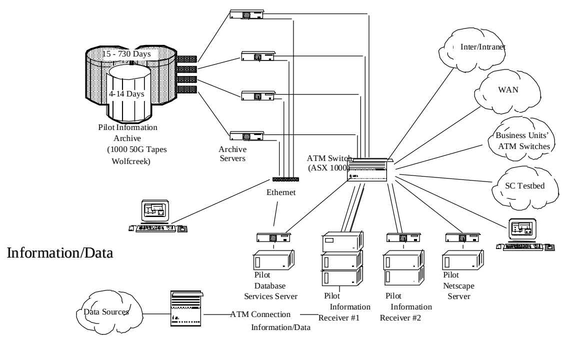


Figure 1. Pilot Hardware: Computer Room

**Desktop Workstation.** The SUN Ultra 2 was selected to meet the functional/operational and performance requirements described earlier. The desktop supports high-resolution display and/or multiple displays using high-resolution monitors coupled with high performance graphical processors. Multiple CPUs are needed to perform analyst operations concurrent with automated processing of data sets. The platform was selected for a high degree of compatibility with COTS software products and a high degree of interoperability with other systems Competitive pricing of the workstation was a significant factor in determining selection. The workstation supports common desktop environment (CDE) and standards including X11R5, TCP/IP, NFS, and SNMP. It is compatible with the Solaris operating system. The next figure depicts a typical user site configuration.

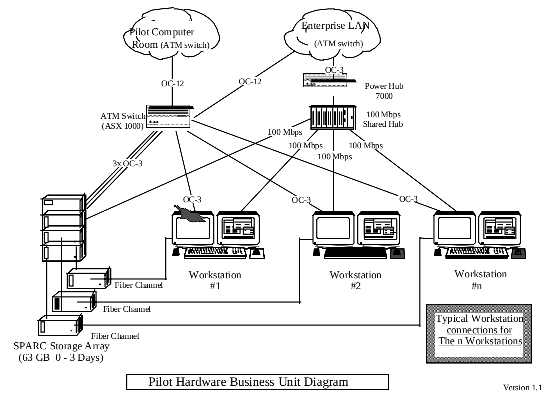


Figure 2. Pilot Hardware: User Site

**Server.** The system uses archive servers (SUN Ultra Enterprise 4000), image servers (SUN Enterprise 3000), Netscape servers (SUN Ultra E 2), database servers (SUN Ultra E 2, later Ultra Enterprise 3000), and workgroup servers (SUN Ultra E4000). The servers support high I/O throughput for RAID disk and high performance tape drives. High transaction processing capability was required to meet the needs of large numbers of system users. Redundant components are incorporated to ensure a high degree of fault tolerance to support mission critical operations. The servers were designed for expandability, so that the same hardware could support increased system use and could be upgraded for enhanced operations. The servers also were designed for flexibility in network connectivity, allowing for the use of a variety of network connections, including asynchronous transfer mode (ATM).

**Mass Storage Subsystem.** The robotic tape drive library (StorageTek Wolfcreek library using SD-3 Redwood magnetic tape drives) supports the archiving of large amounts of data and the concurrent access/retrieval of previously stored data by multiple users. To do this, the mass storage subsystem met performance and capacity requirements for aggregate data transfer bandwidth, storage capacity, low latency for retrieval of archived data, high degree of reliability, and ease of maintenance.

**Network.** The network consists of three Fore ASX-1000 ATM switches connected by OC-12 fiber optic cable. Archive, database, and Netscape servers are attached to the ATM switch via OC-3. Pilot workstations are connected to the ATM switch for data transfer and are connected to a shared 100 Mbps hub for command and control functions.

**Architecture (Software)**

The logical architecture of the software consists of five areas: the user interface, the distributed information management (DIM) engine, database functions, mass storage functions, and the thumbnail and overview creation program.

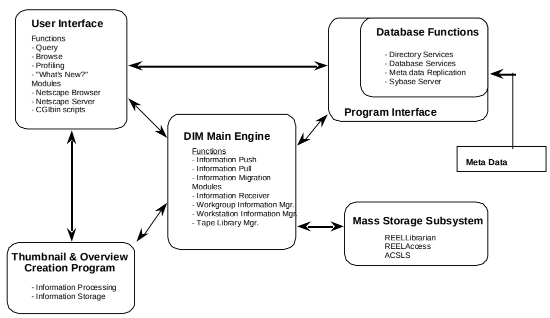


Figure 3. Pilot Software: Logical Architecture

**User Interface.** The user interface supports querying of images, browsing of thumbnails, and allows the user to create profiles in which the user specifies a list of image attributes and the system lists corresponding images and their attributes. The user can then select images for retrieval. The user can also stipulate whether particular profiles can be shared with other users or not and list the users with whom the profile can be shared. Upon retrieval, images can be loaded into several other tools for analysis. The majority of these functions can be invoked from a Netscape browser, the users' primary interface, which has been customized to their needs. More advanced analysis tools can be selected from the common desktop environment (CDE) main desktop.

**Distributed Information Manager (DIM) Main Engine.** This application was the core of the software development effort. C++ and Java were used in the development. Instantiations of this engine were running on the workgroup servers, workstations, and the image receive servers. This software is responsible for image push, image pull, and image migration. The DIM is also responsible for the management of storage on local file systems. Given a specific file system to manage, the DIM monitors usage against low and high watermarks and will purge files as space is needed. This function is performed through combining three factors: total time since entry into the system, last re-use, and size of the file. These three factors are given relative weights and appropriate files are purged when the high water mark is reached in the file system. The image push process notifies the workstation image manager upon receipt of new data. The workstation image manager then contacts the workgroup image manager to retrieve the appropriate images or data sets of interest. The data is simultaneously stored to intermediate tertiary storage (tape library) and the database is updated with directory information.

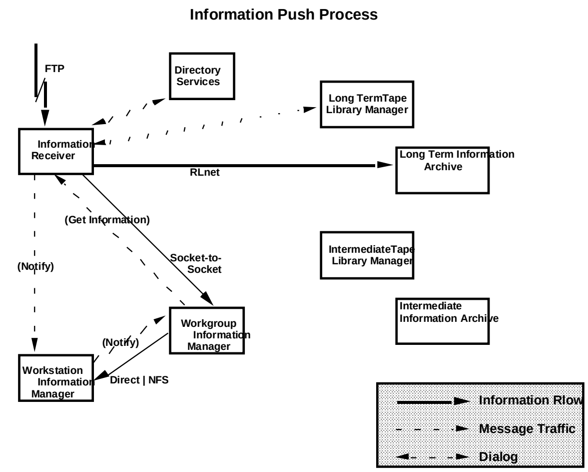


Figure 4. Information Push Process

In the image pull process the workstation image manager checks the database for appropriate attributes and location information, and notifies the workgroup image manager which retrieves the image from one of the following: another workgroup image manager, intermediate image archive, long-term image archive, or the image receiver.

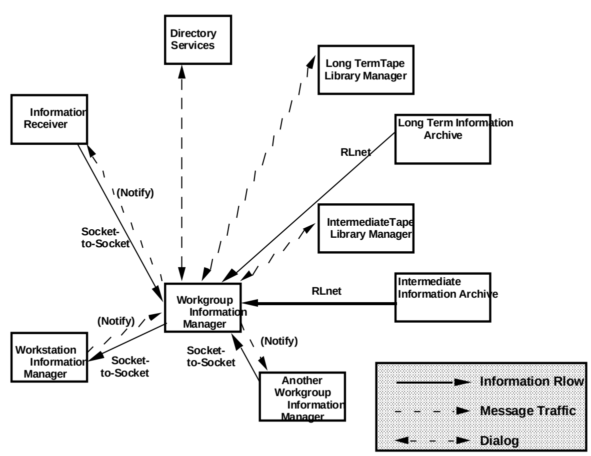


Figure 5. Information Pull Process

In the image migration process, the workgroup image manager automatically moves data to/or from intermediate and long-term image archives as necessary, based upon the age of the file. This software is a combination of developed software and integrated COTS software products.

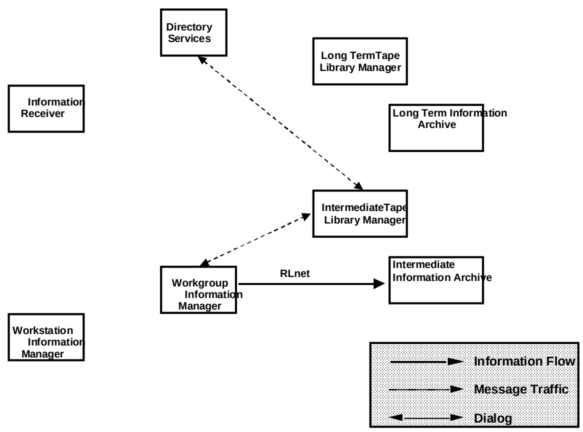


Figure 6. Information Migration Process

**Database Functions.** The database functions comprise directory services, database services, and metadata management. Directory services consist of information about the physical location and physical attributes of the data set. The database contains the location of all data sets including duplicates that may reside on several workgroup servers and tertiary storage simultaneously. This allows the DIM to access data sets from the most efficient location and offers fault tolerance through availability of redundant copies of data sets.

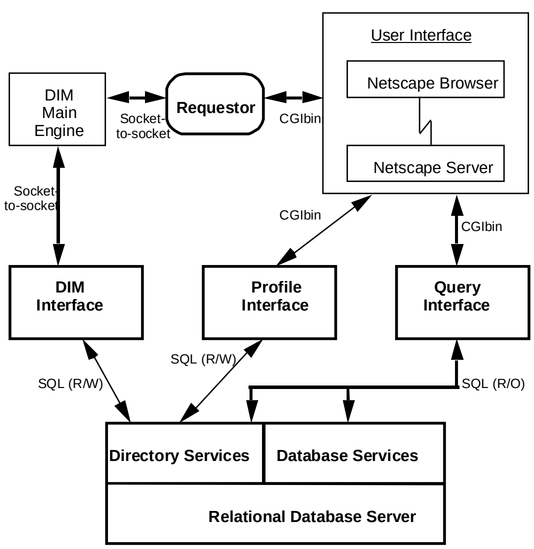


Figure 7. Anatomy of Database Subsystem

Figure 7 shows the key elements of the database subsystem as well as the interfaces to the DIM application and the user interface system. As described earlier, the DIM is a distributed application that resides on every host on the network that stores or processes images. The user interface is composed of an HTML server (Netscape Commerce) and web browsers (Netscape) on each of the user workstations. The relational database is implemented in Sybase and is logically divided into two pieces.

- Database services contains metadata about images and is either imported from external source or is derived from the image header.
- Directory services contain records of all images and all instances of images in the system.

Each of the other pieces is a bit of "glueware" that allows the three major pieces to interact. These pieces are the following.

- The requester, which, when a user requests an image, passes a message into DIM that triggers the transfer of the image.
- The DIM interface, which allows the DIM to find all possible instantiations of an image so that it can select the optimal one.
- The profile interface, which allows users to cause new incoming images to be routed to them on receipt.
- The query interface, which allows users to peruse the archive and select the images they want to access.

**Mass Storage Functions.** The mass storage functions support two requirements: intermediate and long-term image storage. Files up to four days old are stored on-line in RAID file systems of the workgroup servers. Files up to two weeks old are stored in intermediate storage, 10 GB tapes in the library. After two weeks files are moved to long-term storage, 50 GB tapes in the library, where they may remain for up to two years. The mass storage COTS software products are StorageTek's REELlibrarian and REELaccess for I/O and media content management and ACSLS for volume management.

**Thumbnail and Overview Creation Program.** This program is responsible for image processing. Thumbnails are created at the time the image is received and reduced resolution data sets are created based on profile and query requirements. The thumbnails provide a quick viewing capability to the analyst. Reduced resolution sets are generated by COTS Electronic Light Table package.

## Selection and Evaluation of Key Software and Hardware Components

The hardware and software described below compose the mass storage component of the system. Selection criteria for hardware included customer preference, historical relationships, cost, scalability, capacity, availability of support, and compatibility with legacy applications. Selection criteria for software included functional capabilities, compatibility with hardware and software products, cost, maturity of the product, and availability of support.

**Storage Management Hardware**

A StorageTek Wolfcreek library was selected for use by the system based on capacity (approximately 40 TB combining 50 GB and 10 GB media), price, and performance. The STK SD3 tape drive was selected due to the combination of a high-density cartridge (50 GB) to be used for long-term storage and a low-density cartridge (10 GB), which facilitated faster access to data, to be used for intermediate storage. Evaluation criteria included, library capacity, sustained drive transfer rate overall performance, and average access time to data. The high capacity of the media and overall library cost were the dominant factors in the selection of the STK Wolfcreek library using the SD3 drives.

**Storage Management Software**

The software products selected were REELlibrarian, REELaccess, and ACSLS from StorageTek. REELlibrarian offers centralized media management capabilities and is used to track and control user access to data. It is capable of implementing media rotation policies, tracking files, and automatically recording each file written to a volume set. Among its key features are network-wide access to tapes and tape drives, which was important in the proposed distributed environment. Other key features were the programmers' command line access, C-function library, and API, which enable programmers to integrate REELlibrarian's capabilities into custom applications. Its support of multiple density drives was consistent with the selection of the SD3 tape drive.

REELaccess was selected to provide clients in the network direct access to tape drives in the library. This facilitates successful sharing of a library across several systems.

The ACSLS software was required to enable automated volume management control of the STK Wolfcreek library.

Evaluation criteria included overall performance, granularity of control, and ease of configuration, compatibility with specified hardware, and maintainability.

The dominant factor in selection of the REEL product was compatibility with hardware and a very fine degree of control that the product offered. The product allowed the user to allocate drives for specific functions. This was important because of the requirement to ensure no loss of data.

**Database Software**

The COTS database software product selected was Sybase. The customer selected this product because of its compatibility with systems already in place, they currently had a site license (price), and related systems that interface with the pilot program.

## Pilot Implementation Issues

The first step in implementing the pilot program -- after hardware and software selection and evaluation -- was to develop the customized software such as the DIM main engine and related components. This step was followed by integration of the developed components with the selected COTS products. Finally, operational issues could be addressed after initial implementation.

**Development**

TRW's primary software development effort was directed at the DIM main engine, which is composed of five modules: image receiver, workgroup image manager, workstation image manager, interface library manager, and image feed manager. Development languages were C++ and Java as described earlier. The pilot team was comprised of six developers who were responsible for development, software maintenance, and enhancements.

**Integration**

The integration process revealed several problems related to both hardware and software products, including the following.

Interface problems with the DIM software using the API of the REEL products, such as undocumented responses and unexpected behavior. The developers and integrators of the REEL products perceived them to be immature and difficult to embed in the pilot applications. For example, the COTS controls were not set to store records as large as one GB and would therefor not handle the end of tape properly.

- Problems with COTS software products not meeting all performance requirements when integrated with the hardware. For example, to get the sustained transfer rate advertisement in the literature, one must have a direct feed from cache. Tape drive performance problems were traced to the operation of the tape drive transport. The specific problem was poor performance in terms of transfer rate while writing to the media. The original configuration was the attachment of two SD3 tape drives for single E4000 server. The write operation yielded a transfer rate of approximately than 50% of the vendor specification. Analysis of this problem revealed that the socket connection from the REELlibrarian client product was not streaming the data into the drive at an acceptable rate**.** This low input rate caused the drive to empty its buffer to tape and de-tension the transport during the write process and further degrade performance. These problems were addressed by adding individual tape slave servers connected to single tape drive. A SUN Ultra E2 was selected with the REELlibrarian client software as the primary application. This increased performance by ensuring that the client software had the necessary system memory available for streaming data into the buffer of the tape drive. The modifications to the hardware configuration allowed for the operation of the drive at approximately 80% of it specified capability.

These problems allowed a general evaluation of vendor responsiveness to product problems. We found that as you used the vendor product in environments different from standard IT environments, it was difficult to get the design engineer assigned to investigate the problem and the maintenance engineers did not understand the internals of the control software well enough to diagnose and fix the problem. Vendors from Storagetek worked closely with the TRW team to attempt to duplicate problems. A major difficulty included reproducing or isolating all the system conditions that caused specific problems.

## Operational Experiences

**Maintenance and User Interaction**. The operation and maintenance of the system used the same collaboration tools as the users. Web pages were maintained that reported the latest system upgrades and reported system problems. A weekly user group meeting provided feedback to the development and support team about a range of issues including user preferences, hardware and software anomalies, and desired functionality. Six engineers were responsible 24 hour support of eleven locations. As a problem was reported (via e-mail or pager) a TRW engineer would respond to observe the problem as it was happening. This offered insight into the actual use of the system and allowed a detailed first hand analysis to be posted on a Web page for further investigation by a senior developer.

**Configuration of Servers and Workstations.** In the beginning of the deployment effort, it took two weeks to configure a server with all the COTS and GOTS products and approximately three days to configure a client. By configuring a jumpstart process, we were able to configure new servers (as many as needed) in 4 to 6 hours and the new clients in one hour. Using the jumpstart process, all machines could be configured simultaneously over the network.

New versions of COTS products were tested in the test laboratory and once they past the acceptance testing were rolled out to all equipment during non-peak use times. One hour a week was reserved for upgrades and modifications and this time slot allowed for most maintenance to be accomplished without interference with the operational users.

**Operating System Upgrades**. Significant operational and support issues arose during the process of evaluating and testing operating system upgrades of workstations and servers from Solaris 2.5.1 to Solaris 2.6. The effort required approximately 4 months to complete testing. Issues requiring attention included the following.

- Minor modifications to and anomalies in the desktop CDE were noted.
- Several critical evaluation tools were unsupported under Solaris 2.6.
- Migration instructions for some software products were incomplete.
- Files larger than 2 GB caused file system errors when mounted on machines with Solaris 2.5.1.
- A new FORE ATM driver was required with Solaris 2.6 when the workstation was configured with multiple ATM ports.

Although the majority of these issues could be resolved by applying operating system patches or workarounds, the recommendation was to delay upgrading the operating system.

**Scalability.** Operational issues also arose when scalability was tested. When the prototype started operations the three access timelines were examined. While the architecture met the requirements of the customer, the difference in access times from the two tape media (10 GB and 50 GB) was not as significant as originally anticipated. As production of the system increased, the value of the higher capacity media cartridge was more significant than the difference in access times between the two media types. All 10 GB cartridges were replaced and the operation system could retrieve data in approximately five to six minutes from the high capacity media.

Increasing the number of workstations from the original 25 also increased the amount of imagery to be processed, stored, and retrieved. This strained image processing and significantly increased response time for image retrieval and display. TRW addressed these issues by rerouting archived data from the image receiver to the workgroup servers, modifying archival procedures, and implementing a circular long-term storage strategy. This replaced the requirement of retaining images for a period of two years and reduced the complexity of the storage management. Verification procedures for these modifications were established and the solutions were implemented without disrupting system operation.

## Lessons Learned

The pilot implementation yielded several lessons learned, from which the following critical issues were identified.

- Complex systems can be successfully developed and supported by integrating large numbers of COTS/GOTS products as demonstrated by eighteen-month schedule and completion of the pilot program.
- Strict adherence to open standards is needed to facilitate platform independence and substitution or replacement of underlying system components.
- Coordinate the configuration of COTS/GOTS components to ensure compatibility.
- Develop strong relationships between vendors of key system components to ensure smooth scalability and continued performance.
- Evaluate products continuously for risk mitigation purposes to address problems such as vanishing vendors, product release incompatibilities such as OS product lag, modified features, deleted functions, etc.
- Periodically evaluate internally developed software products against COTS products. Reduction in custom software development and maintenance costs may be achieved by the replacement of internally developed products.

## Internal Research and Development

### Background

The Common Information Infrastructure (CII) Lab was conceived in 1998 as a center for cooperative research in information technologies. The overall objective of the laboratory is to study and develop high performance system architectures to distribute process and manage large amounts of data for a variety of applications. TRW is currently establishing relationships with academic institutions and vendors to establish collaborative research efforts.

The purpose of the CII Lab is as follows.

- Provide an environment for performance modeling, benchmarking, and simulation work.
- Showcase vendor solutions in a working environment.
- Assist vendors by providing feedback about their products and performance results from various test configurations.
- Provide objective testing of similar products and configurations.

The CII Lab's five main objectives are to evaluate current mass storage subsystems, evaluate ORB products, develop a model of a real-time software framework, evaluate various UCA-compatible hardware and software configurations, and give the customer and participating vendors the opportunity to provide feedback. The CII Lab will provide the customer with published findings.

### Research Areas

**Unified Cryptologic Architecture.** Department of Defense (DoD) information technology management is developing an architecture that will allow them to share information and products. There is a concerted effort to integrate their architectures into a common architecture, called the Unified Cryptologic Architecture (UCA), so that they can perform their missions through collaboration. The UCA will rely on distributed data processing as well as distributed data and metadata storage. Incoming data will be handled by generating metadata to describe the collected data, the processing steps that are required to correctly format the data, the location of the data, as well as other data-related information. User data requests, which are metadata based, will be handled by locating all relevant data, performing data processing (if necessary), and providing the requested data to the user. The requested data may be stored locally or remotely, and may also be processed locally or remotely.

Performance requirements for UCA-compliant systems include real-time data collection, data transformation, analysis, and information dissemination.

**Mass Storage.** An important element of this process is how the data and metadata are transferred and stored. With the amount of information collected, data will be stored locally and remotely. Distributed mass storage subsystems will be a critical component for systems that meet UCA standards.

The comparison of similar hardware and software configurations will provide the customer with beneficial insights on the quantitative aspects of performance and the qualitative aspects of system configuration, integration of various hardware and software components and their operational characteristics

**Medical Information Management and Image Analysis.** Image analysis in the areas of medical fields such as MRI, CT, and radiology will be addressed as a long-term goal. The Pilot program baseline will be used to develop systems that can collect and analyze medical imagery and provide collaborative tools for information dissemination and research activities in this field.

**Software Frameworks.** In order to build the future UCA-compatible system it will be necessary to provide a framework that can operate in real time for high data rate signals. The current implementation of the real-time framework uses Information Objects in a transaction-based environment. This implementation is unlikely to support a high data rate environment. The CII will evaluate other framework options more conducive to high data rate processing of real-time data.

**Current Status**

TRW has commitments from over 20 vendors of hardware and software to participate in this effort. Relationships with George Mason University and University of Maryland are being pursued in the areas of benchmarking and modeling/simulation of Mass Storage systems and software frameworks and their components. TRW's Common Information Infrastructure Laboratory will be operational March of 1999. We will update the conference members of our efforts in more detail at the time of the conference.

## Acknowledgments

I would like thank Gail Phipps and Christopher Read of TRW for providing the historical background and technical information about the pilot program. I would like to thank Christine McQuaid and Karen Davis for their editorial assistance.