Joint Library & NSIT Digital Preservation Project Proposal

Introduction

We were tasked to examine ways to preserve certain digital objects, initially artifacts produced by the library and university e-mail, and to propose a joint project between NSIT and the library to implement a digital preservation system for these objects. The project is understood to be highly constrained with regard to funding and staff resources.

Given the resource constraints for this project, we base our recommendations on the following principles:

- Leverage generally available, open source, public domain or open licensed software where possible
- Leverage existing infrastructure and staff skill sets where possible
- Utilize commercial software when it is "affordable" and fits within both the university infrastructure and the accepted principles of digital library preservation

Proposal Summary

We propose that a pilot project be undertaken to create two applications sharing some common infrastructure and technologies. One will provide e-mail archival and the other will be used for preservation of digital library objects. The goals of the pilot project would be to better estimate resource requirements for a generally available e-mail archival service and to validate usability and functionality requirements prior to full production. Since these requirements are better understood for library digital objects, the pilot system would be quite nearly production-ready for the library's purposes.

The applications will make substantial use of work done elsewhere with some local programming to integrate the various components and layers. The database, content storage and storage management infrastructures would be provided by NSIT. Appendix B provides details regarding the components, their origin, and any associated costs.

The library artifacts would be stored in a "dark archive" while e-mail would not. A dark archive is one that preserves information but does not necessarily allow for real time access by anyone other than archive staff. We recommend that the library provide a programmer/analyst with specific skills who would develop the mechanisms for packaging library artifacts for submission to the dark archive. The dark archive itself would be implemented using software developed at the Florida Center for Library Automation (FCLA), using NSIT's Storage Area Network (SAN) and IBM Tivoli Storage Manager (ITSM) systems for the back-end. The FCLA software appears to be the only such available that was expressly developed for purposes such as ours.

We propose that a pilot e-mail archive service be created using locally developed programs to move e-mail from special IMAP folders established for registered users. The processes are described in detail in this document, and proof of concept code has been written and tested to validate the design behind the proposal. The e-mail archive would use parts of the same storage infrastructure established for the library objects; however, we do not believe that a dark archive would be acceptable for an e-mail archive. Some development effort will have to be made to allow users to search within their own archived e-mails and to retrieve messages from the archive.

The size of the e-mail archival pilot community is to be determined. However, we suggest that it be grown in steps and not to exceed a final population of 50 users. Once it is ready to support more than that size population, it should no longer be considered a pilot project.

NSIT would have to contribute some reasonable staff time for database development and administration, as well as storage and server management. Additionally, NSIT would provide programming assistance to aid in the production transition for prototype code, creation of the search interface for e-mail, integration of components, and possibly some Java skills to assist with installation and configuration of the FCLA software. However, some of the staff requirements could be outsourced, for example to FCLA staff.

The hardware and software costs for the pilot projects should range between $25,000 and $35,000. All of that investment would carry over to a full-scale production environment if such becomes reality.

Joint Library & NSIT Digital Preservation Project Proposal

Policy Issues

No archive service at the university should be viewed as a form of alternative, extended data storage. If objects are easily archived and retrieved, then there may be a temptation to view an apparently free service as a form of free storage. The archive is being developed and maintained for the three purposes for which archives exist (legal and regulatory compliance, to improve operational effectiveness, and historical preservation). Any other use needs to be treated as an abuse of university resources.

This raises the question of whether to implement billing for the e-mail archival service (as well as how to fund and account for any future archiving efforts). Implementing an FAS-based authorization and billing system for this purpose would substantially increase the effort required. One could also ask "What does it mean to bill for information that may be stored for decades?" We propose that the pilot project should not include the development of charging mechanisms.

There is also the question of eligibility for the service. Should the service be restricted to staff, faculty, or subsets thereof? Are there legitimate student uses for e-mail archive that the university would be willing to support?

Finally there is a question of whether or not access to e-mails should be granted to anyone other than the sender or recipient of the message. In particular, is there a level of authorization to view archived e-mails that would be granted to someone such as the university archivist or some officer of the university?

We look forward to an opportunity to discuss these issues with the appropriate parties.

General Archival Flow

The components of an archival system are depicted in Appendix A. In the simplest terms, archival can be seen as a process that looks like:

Object Production->Object Ingestion->Storage and Ongoing Management

Object Production includes preparation of a Submission Information Package (SIP). In the pilot project, this is fairly simplified. Library objects are already being produced under the aegis of the Digital Library Development Center. SIP preparation functionality would be provided by a proposed Library programmer. For e-mail, we would forego SIP preparation as the message itself would constitute the SIP. The SIP is transmitted to Object Ingestion as a METS (Metadata Encoding and Transmission Standard) object.

Object Ingestion includes Content Validation, Metadata Harvesting, and preparation of an Archival Information package (AIP). The AIP is then sent to storage management services for ongoing, long-term operational support.

Storage and Ongoing management includes providing a Metadata Repository, Hierarchical Storage Management, Media Management, Virus Checking, Fixity Checking, Disaster Recovery Management, and Format Migration.

The above processes are sufficient for a dark archive. To open up a dark archive to a more general user population requires additional Access Services. These would include Discovery, Access Control (AuthN + AuthZ), Withdrawal, and ongoing Rights and Permissions Management and Migration.

Finally, there may be external services required such as Authentication and Billing (previously discussed). Any authentication would be based on the university's LDAP service.

Joint Library & NSIT Digital Preservation Project Proposal

E-mail Archival

RYO (Roll Your Own) E-mail Archival

E-mail Submission

We considered the following ways in which a user could "submit" e-mail to the archive.
1) User forwards a message to a special address, e.g. archive@uchicago.edu
2) User has an attribute in LDAP which indicates that all of their e-mail is automatically archived
3) Same as #2, but the attribute is maintained in the archival application database
4) User copies or moves any messages to be archived into a special folder, e.g. "Archive Depository"

We discussed the implementation of these schemes with NSENSA and their support with Tech Line. Our consensus was that the use of #4, the "Archive Depository" folder, was the simplest to support and least problematic to implement. This option would also allow a user to create a filter through Webmail to automate movement of certain e-mail into the archive.

E-mail Ingestion

Using a combination of zLinux, Perl, and MySQL, we created a "proof of concept" program to implement the "Mailbox Processing Logic" described in Appendix C.

In short, the program:
- Loops over the rows from a table of registered users in the e-mail archive application's database
- Reads the messages from the users' "Archive Depository" folder
- Extracts metadata from the messages
- Matches mail addresses against LDAP entries to create a list of CNetIDs authorized to view the message
- Stores the metadata and LDAP information into the application's database
- Stores the original message and an XMTP form (an XML form of the MIME objects) into filesystem files. This includes attachments, since they are MIME objects within the e-mail.
- Deletes the processed messages from the "Archive Depository" folder

To aid in subsequent searches of the archive, an additional step would be necessary to update an index of the user's e-mail. Each user in the system would have an index of e-mails either originating from their "Archive Depository" folder or in which they were referenced to as an addressee. This could be done with a couple of days effort using a search engine toolkit such as Lucene (or Plucene for Perl).

Aside from the search indexing, this e-mail ingestion code could be made ready for pilot testing with an additional week or so of effort. This would include adding robust error handling and logging, securing the internal password stores, and a code review.

User Registration

The proposed ingestion model requires that users register for the system. A simple web page and backend CGI script to validate LDAP authentication are all that are necessary for registration. The CGI script will check for the existence of the "Archive Depository" folder and create it if necessary. Like the Mailbox Processing logic, this is described in Appendix C. This process was also prototyped using zLinux, Apache, MySQL, and Perl.

If any sort of billing is required for users of this system, then an FAS ledger number would have to be provided and validated. Again, it is our belief that validating FAS numbers and implementing billing mechanisms would greatly complicate the implementation of the system. Therefore, we propose that no billing be done during the pilot phase of the project.

User Access, Search, and Display

It is proposed that access to e-mail be initially based solely on CNetID. Implementation of Role Based Access Control, Grouper/Signet controls, or some other more complex access control scheme can be considered when the appropriate mechanisms exist and are in general use elsewhere at the university.

Whether or not to allow someone such as the University Archivist or specified officers of the university general access to archived e-mails is a policy decision to be made by the appropriate parties.

Users will not be allowed to delete e-mail from the archive. It may be desirable to allow some form of policy based automation to delete e-mail at a later date, but that is not proposed at this time.

Search and display would be enabled through a web interface and search engine toolkit such as Lucene. Lucene would allow the system to index the content of mail messages being stored in flat filesystem files while storing the indices in the application database. Each user of the system would have their own index to messages they either "own" or in which they were referenced by address. The alternative is to store the messages in the database to enable SQL searching, causing the database to grow quite large and complicating most other aspects of the system.

Commercial Software Alternatives

While we did not conduct a formal RFP or RFI process, we did examine a number of commercial e-mail archival systems and found only one that might be considered given our constraints. The majority of commercial systems are designed to work with Microsoft Exchange or Exchange and IBM Lotus Domino. Perhaps only one quarter of the dozen or so legitimate commercial e-mail archival systems work with generic POP or IMAP servers. Additionally, the commercial e-mail archival market is being driven by compliance issues and is not oriented toward the long-term issues of preserving e-mail. Finally, the software costs for commercial packages tend to be quite high, starting well over $100k.

Optical Imaging Technology Inc.'s DocFinity Email Manager is a commercial solution that could be acquired for approximately $21,000 for the server software and a concurrent user cost that would range from approximately $400 to $600 per concurrent user, depending upon the number of licenses acquired. The concurrent user license cost applies to the number of users actively engaged in accessing and retrieving e-mail from the archive, so this could potentially be tightly managed, perhaps arbitrarily limited to two concurrent users.

The disadvantages of this approach are:

- The OIT authentication model would not work well for us without customization
- The OIT Email Manager is part of a larger Integrated Document Management (IDM) suite. As such, the user interface paradigm is more complex than need be for the limited function we envision.
- The data would reside in non-standard, proprietary data stores.

These limitations could be overcome; however, the amount of work necessary to do so would not be very different from the amount of work required to implement the recommended, locally developed solution.

Archival for Library Digital Objects

The library currently has about 1 terabyte of objects stored in NSIT's ITSM systems. These are primarily TIFF files; however, other file formats will certainly be produced in the future. Current plans would call for an additional 600Gb of data being produced each year. These are quite modest requirements, given the current state of storage technology. Even allowing for duplicate copies of tapes, the cost of the media to contain this data is less than $1,000 per year. Even if the library's efforts expand greatly, the cost/technology curve for storage should be able to keep pace with their needs.

So, why expend the effort to create an archiving system for these objects? Aside from their intrinsic value, they provide a well-defined sample with which to work to develop an archival model that can later be extended to support other forms of data such as research data, administrative records, web pages, etc.

Objects already produced by the Library would be packaged into a SIP and transmitted to the FCLA software as a Metadata Encoding and Transmission Standard (METS) object. The FCLA software would store metadata into a relational database and create an AIP for storage into the ITSM managed back-end storage.

Common Infrastructure

Staffing Requirements

Digital archiving software presupposes that data, and the metadata required to maintain that data, are packaged for submission according to a relevant XML-based standard, for example, the Metadata Encoding and Transmission Standard (METS), maintained by the Library of Congress. Therefore, in order to support the University's archiving goals, it is proposed that the Library supply a programmer/analyst with XML and XSLT skills, and a knowledge of the relevant standards, e.g., Dublin Core and METS, to package the data which the Library produces together with the associated metadata for submission into the archive. These skills can also be applied to extract searchable metadata for University email packaged in XMTP format if future integration of the e-mail archive and Library archives is desirable.

NSIT would provide a data architect to assist with finalizing the design for the e-mail metadata store and a DBA to assist with creating both metadata stores. The FCLA data dictionary is already well defined, so no design work is necessary for that. This should only require a couple days of effort.

NSIT would provide some programming assistance for completion of the e-mail ingestion and registration functions (about two person-weeks effort), development of a search and display mechanism for the archived e-mail (perhaps a further two person weeks effort), and support for installation of the FCLA software. Again, the FCLA installation could be outsourced, for example to FCLA staff.

A further staff requirement exists for the development of processes to manage the media controlled by ITSM. This is not just a requirement for archival; however, archival's long-term needs certainly bring it to the fore. The lifespan of ITSM tape media is not currently being managed with regard to age or number of mounts. Processes to manage the life-cycle of ITSM media should be developed regardless of the potential use of ITSM for archival storage. Archival will require that dual copies be maintained and that the media be periodically migrated. When media is migrated, its contents should be validated for fixity.

Technology Infrastructure

The FCLA software from Florida uses IBM Tivoli Storage Manager (ITSM) as its backend storage manager. This fits well with our principle of using existing infrastructure and staff skills. Because University Digital Library objects are already being stored within ITSM, managing them as a true archive would not place any additional burden on ITSM resources. Adding e-mail archival to an existing ITSM server should be possible during the pilot phase of the project. This would give us a baseline for projecting any additional resource requirements for an expanded e-mail archival system.

A Hierarchical Storage Manager (HSM) agent would be required to manage the migration of content from near-line storage into the ITSM managed backend storage. Unfortunately, HSM is not well implemented on Linux as yet. There is a great deal of work being done to address this, but it is far from mature at this point. The alternatives are to base the storage on either AIX or Windows filesystems. Fortunately, the OnStor NAS device just installed by NSIT allows Linux and Windows servers to transparently and simultaneously access the same filesystems. Therefore, we could use Linux to manage the archival processes up to the point where HSM is required, and then turn HSM management over to a Windows HSM client. Some testing is required to verify that this will work as desired.

Joint Library & NSIT Digital Preservation Project Proposal

ITSM compatible HSM agents are currently available from EMC/Legato and Caminosoft. The Caminosoft agent has a list price of $9590, or approximately $7000 with educational discounts. IBM will announce new HSM agents for ITSM in the middle of September. Pricing has not been set for the IBM agents, but it will be based upon the number of terabytes under management. This may prove to be very expensive over time as the data being archived grows.

Aside from the possible HSM issues, the proposed software fits well into a LAMP (Linux, Apache, MySQL, and Perl (or PHP or Python)) model for server requirements. There are some small adjustments to the LAMP model necessary. While MySQL was serviceable for proof of concept, we recommend using NSIT supported Oracle databases for the metadata repositories. Also, some Java code will be necessary in addition to the Perl. That said, small Linux servers should be perfectly suitable to support this application. zLinux was used for proof of concept and could continue to be used for the pilot phase, requiring no additional server hardware or software be purchased. Two Linux servers (virtual or otherwise) should be sufficient for the pilot project.

Because of the HSM issues, a Windows server would also be required for Windows HSM. If Windows HSM proves unsuitable during testing, then an AIX based solution may be required. A small AIX rack-mount server costs about $8,000 with 3 years of maintenance and support. Alternatively, a BladeCenter with AIX blades (that could also run Linux) could be considered.

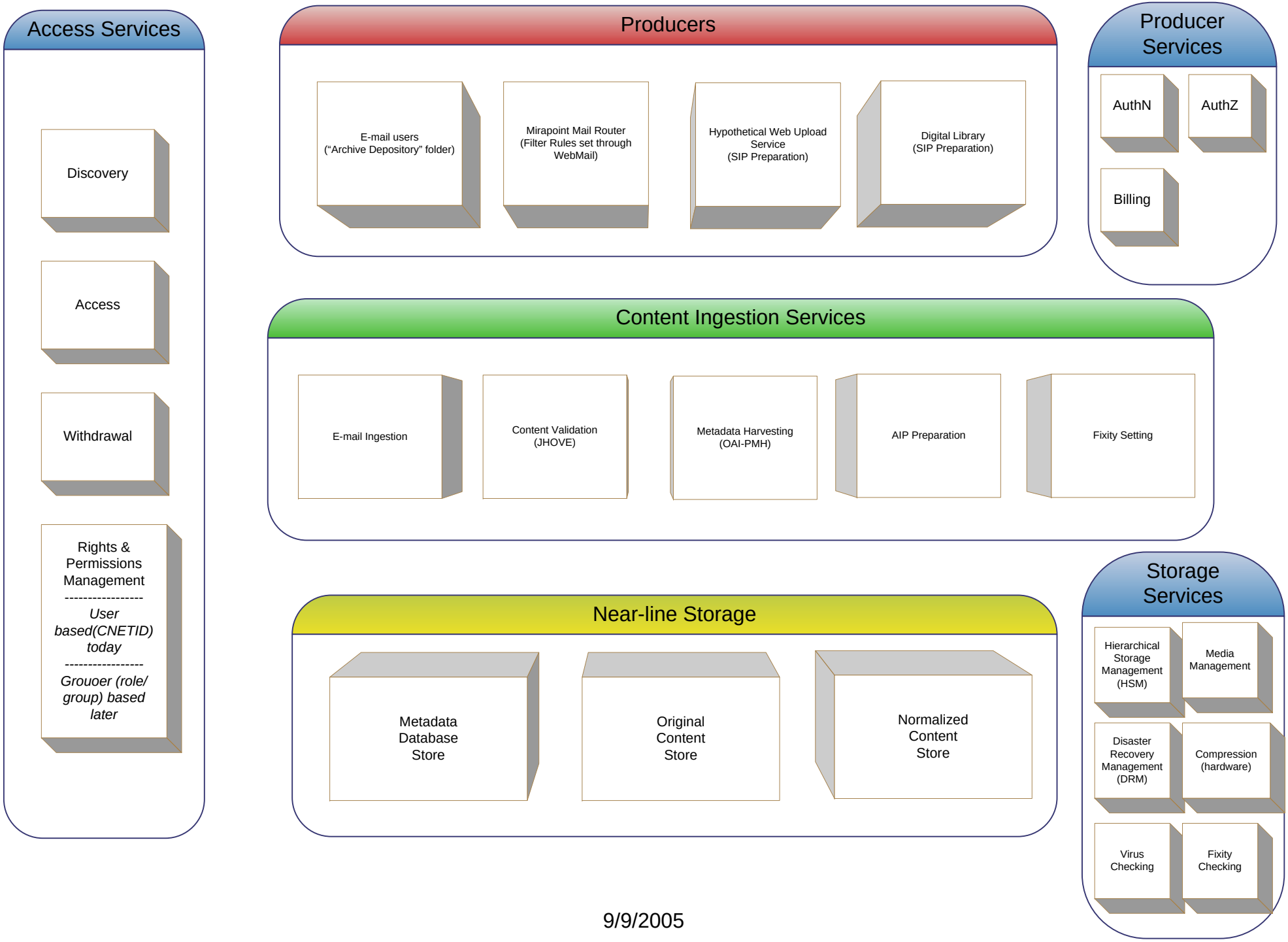
The databases could either be hosted on the Shared Oracle Server or on a zLinux Oracle server. No additional database server should need to be acquired.

Some additional SAN disk storage would be required to hold the metadata databases and the near-line content stores. One terabyte of SATA storage should be sufficient for both the e-mail and library pilots at a fully loaded, NSIT cost of $9,250.

A full production e-mail archival system would likely require additional disk storage as well. The two factors governing the size of the disk pool are the volume of data being archived and the period of time that we wish to keep it on-line before migration to tape. A goal of the pilot project would be to better determine our expectations for those factors.

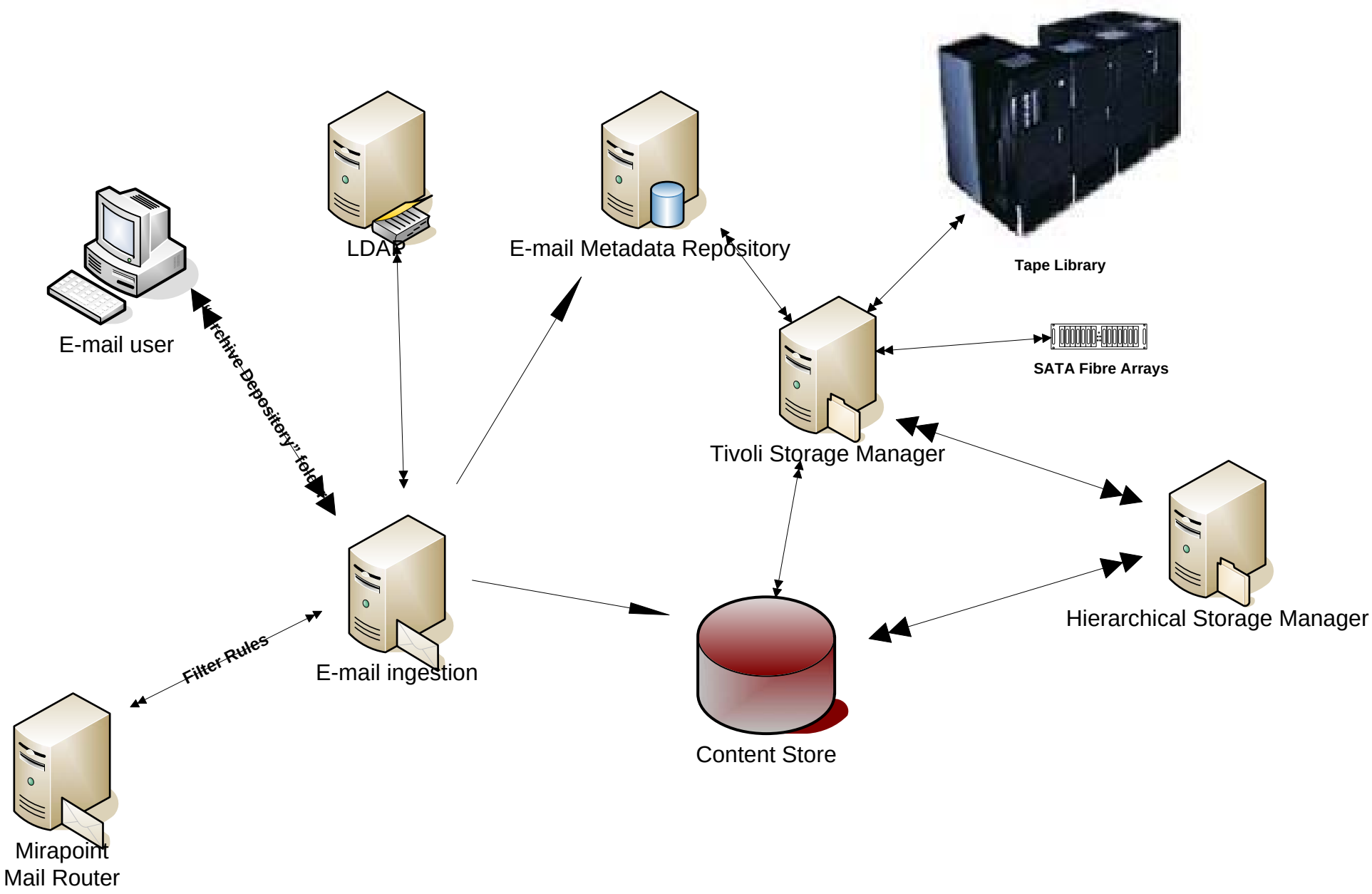
If a production archival service becomes a reality and additional ingestion engines are established for other data sources such as administrative or research data, then it is likely that additional tape drives will be required for ITSM. These cost approximately $15,000 each, and we should anticipate needing two tape drives ($30,000).

Appendix A - Logical Component Model

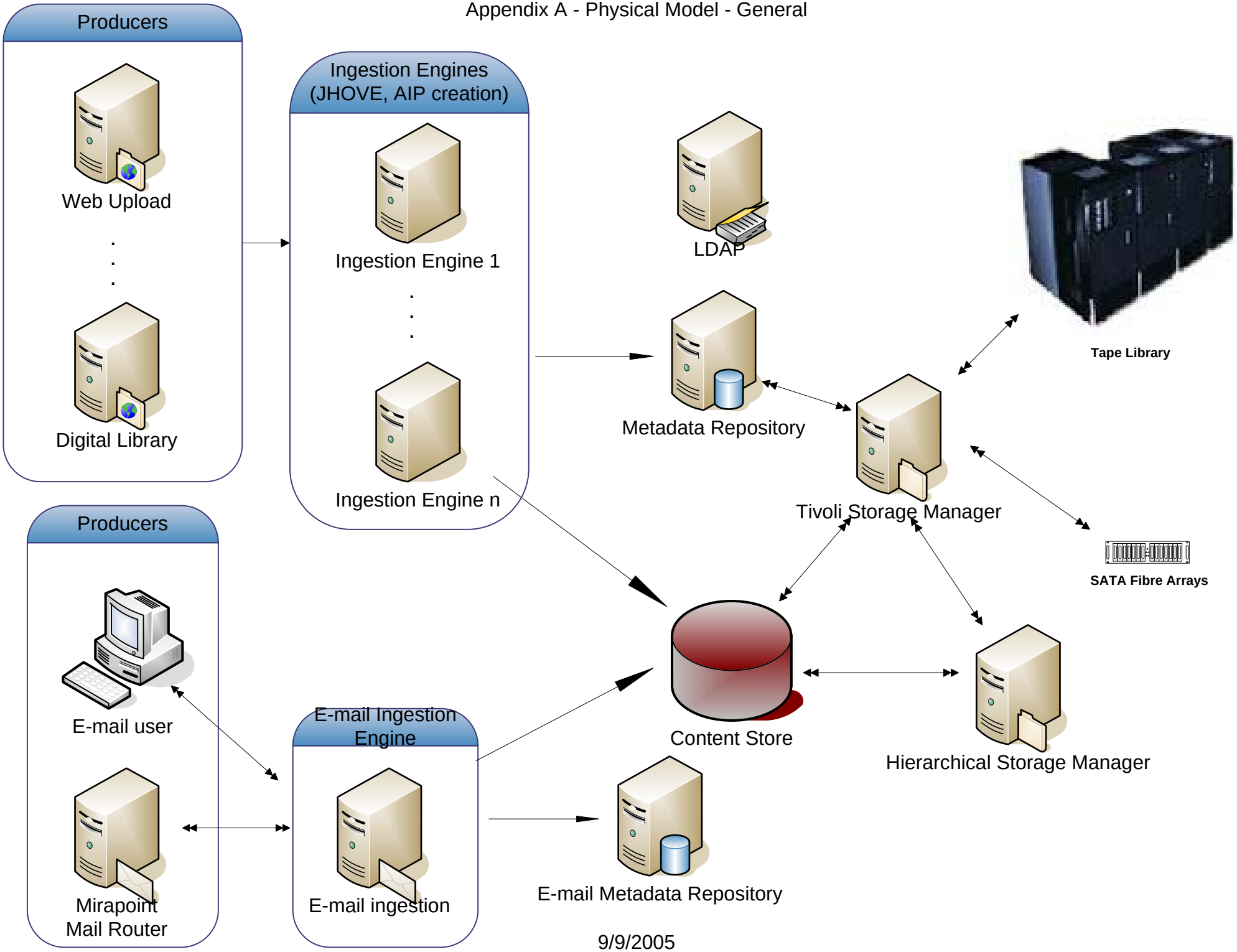


9/9/2005

Appendix A - Physical Model - Mail



Appendix A - Physical Model - General

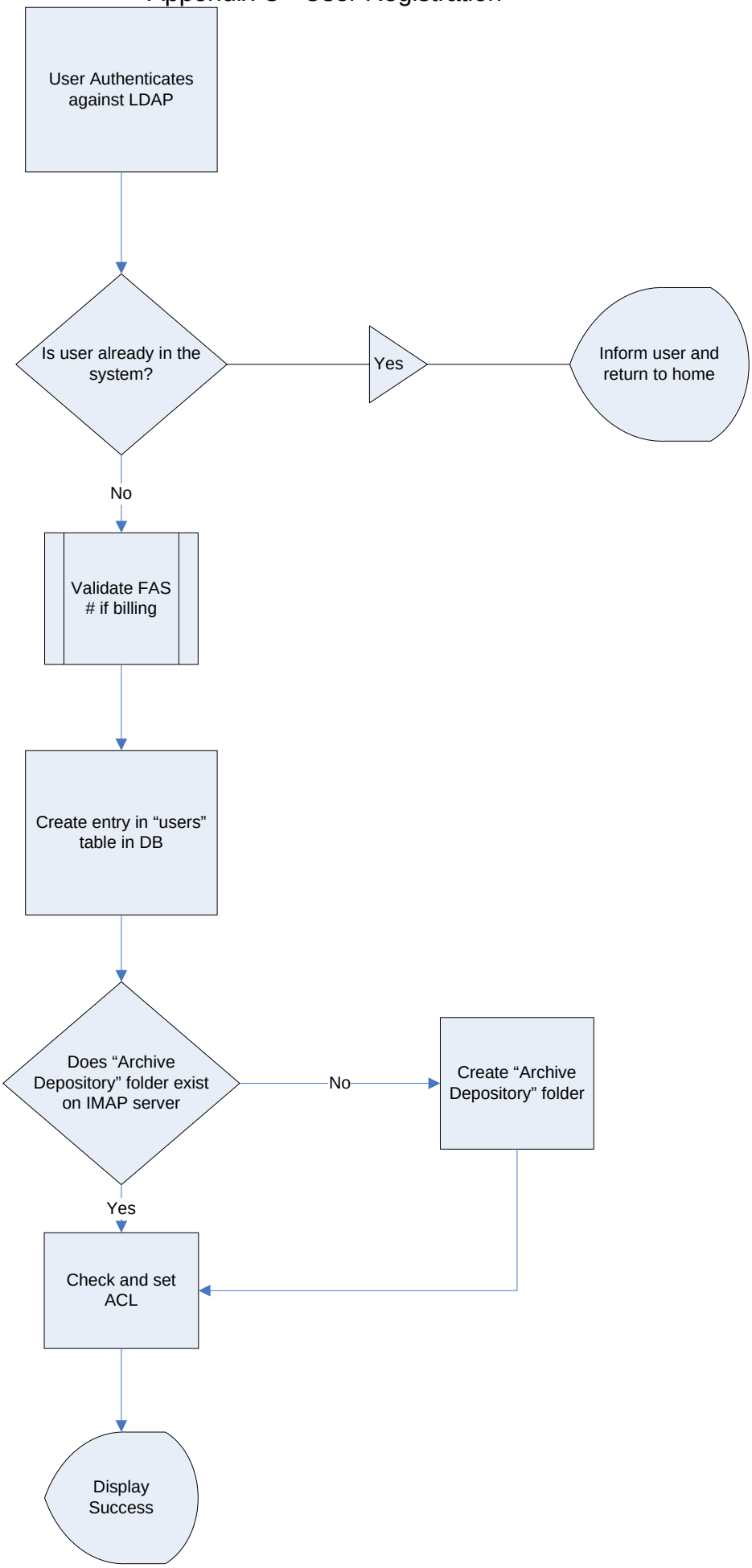


9/9/2005

Joint Library & NSIT Digital Preservation Project Proposal
Appendix B – Component Breakdown

Function	Source or other Notes	Cost or Time Estimate
E-mail Content Production	End users	Already happening
Digital Library Content Production	Digital Library Development Center	Already happening
E-mail Ingestion	Locally developed service	Prototyped, production would require another two person-weeks effort
Digital Library Content Ingestion	Locally developed service	Library supplied staff
Content Validation	JHOVE (Harvard)	No cost
Metadata Harvesting (E-mail)	Part of the local ingestion script	n/a
Metadata Harvesting (Library)	Part of the locally developed ingestion service	No cost
Fixity setting	Standard Perl Digest module	Done for e-mail prototype; ¼ day for library objects
Metadata Databases	NSIT Oracle (either on SOS or zLinux)	Two days each for e-mail and library
Content Store	Virtual NAS server on NSIT SAN	SAN SATA Cost for 1TB; $9,250
E-mail Archive Search & Display	Locally developed service	Would require approx. two person-weeks effort
Hierarchical Storage Management	Caminosoft HSM	~$7,000
Media management (duplication, validation, migration)	Processes need to be defined and developed	TBD
Disaster Recovery Management	Initially consists of making duplicate copies of WORM tapes for off-site storage	Nominal cost to add a handful of tapes to existing off-site storage
Compression	n/a – performed by tape hardware	n/a
Virus checking	n/a for library objects and e-mail is pre-scanned	n/a
Fixity checking	Locally developed script to periodically check digest codes	Two days to integrate
WORM tapes (each holds approximately 1TB of e-mail with compression)	Initially acquire 20 tapes at approximately $150 each	~$3,000
zLinux servers or AIX if using native HSM		No cost for zLinux; ~$8,000 for an entry level AIX server
Windows server for HSM		Approximately $2,000

Appendix C - User Registration



Appendix C - Mailbox Processing