4/27/2010

|  |
| --- |
| The Raging Librarians | Matt Schwartz, Russell Sharpe and Nick Williams  *Presented to: Ronee Francis, Digital Collections Archivist, MLIS,DiscoverArchive* |

# Home_Photo_books.jpg

|  |  |
| --- | --- |
| Vanderbilt University Engm 273 | DiscoverArchive System Engineering Management Project |

# Executive Summary

DiscoverArchive within Vanderbilt University’s Library system currently exists in a very basic implementation that does not necessarily encourage the fundamental purpose of the archive. This purpose is to gather and publish to a common, openly accessible online database any and all faculty and student generated research produced at Vanderbilt University. The intent of this database is to allow anyone free access to the knowledge produced in this academic and research driven environment in an effort to further the combined knowledge of the academic world.

While the aims of the system are in accordance with the founding principles of the open access movement, the reality is that the current implementation of the system is far too rudimentary to handle the volume of data that would be generated by a University wide mandate to publish to the database. In its current state, the archive relies almost entirely on the efforts of Ronee Francis, Manager of Digital Archives at Vanderbilt, to go out and search for research documents and professors to include in the archive. Once she finds a viable document, she contacts both author and publisher to obtain permissions to add it to the archive, and if those are granted she fills out a non-exclusive Vanderbilt license that is uploaded to the archive along with the PDF of the document. If documents do not exist in digital form they must first be scanned by library staff. It is easy to see that this existing methodology can only handle so much data because of the amount of time that the few people involved must spend preparing each document.

To address the needs of the DiscoverArchive system, we worked with Ms. Ronee Francis to determine the requirements of a new system, and the operational scenarios in which it must operate. From there we developed two systems, one with incremental implications and one which with a much more progressive approach. The incremental system builds off of current requirements but adds functions which help automate the system in target areas. The progressive system begins with newly conceptualized requirements and operational scenarios, namely operating under the passage of an open access mandate, and uses a newly designed interface to meet these requirements. Ultimately, based on a presentation and discussion with the DiscoverArchive team, we came to the conclusion that the incremental system is the best option moving forward. However, if the open access mandate is passed, there will be an increase in university funding which will be sufficient to pursue the progressive plan. For this reason the systems engineering design of the progressive system should be kept close at hand since this passage is inevatible, but not in the near future.

Contents

[Executive Summary 2](#_Toc260196929)

[Part I. Analysis of the Current System 4](#_Toc260196930)

[1. Introduction and Background- System Context 4](#_Toc260196931)

[1.1 Unit’s underlying mission 4](#_Toc260196932)

[1.2 Unit current status 4](#_Toc260196933)

[1.3 Unit priorities 5](#_Toc260196934)

[1.4 Unit’s core competencies 5](#_Toc260196935)

[1.5 Role of project 5](#_Toc260196936)

[2. General Description of System Architecture 6](#_Toc260196937)

[2.1 Operational Concept 6](#_Toc260196938)

[2.2 Stakeholder identification and roles 6](#_Toc260196939)

[2.3 External Systems Diagram 7](#_Toc260196940)

[2.4 Identification of Comparable Systems for Benchmarking 8](#_Toc260196941)

[2.4.1 Benchmarking partner criteria 8](#_Toc260196942)

[2.4.2 Benchmarking partner identification 9](#_Toc260196943)

[2.4.3 Benchmarking partner agreement 9](#_Toc260196944)

[3. Requirements Analysis 10](#_Toc260196945)

[3.1 Operational Scenarios 10](#_Toc260196946)

[3.2 Operational Scenarios in Full 10](#_Toc260196947)

[3.3 Input/Output (I/O) Trace 12](#_Toc260196948)

[3.4 Complete Requirements Hierarchy 13](#_Toc260196949)

[3.5 Technical Performance Metrics and Target Levels 13](#_Toc260196950)

[3.6 Requirements Priority Tree 13](#_Toc260196951)

[4 Functional Architecture 16](#_Toc260196952)

[4.1 System Functional Diagram 16](#_Toc260196953)

[4.2 Functional Flow Block Diagram 17](#_Toc260196954)

[4.3 Enhanced Functional Flow Block Diagram 20](#_Toc260196955)

[5 Physical and Operational Architecture 23](#_Toc260196956)

[5.1 General Architecture 23](#_Toc260196957)

[5.2 Operational Architecture 24](#_Toc260196958)

[6 Current System Performance 25](#_Toc260196959)

[6.1 Technical Performance Measures 25](#_Toc260196960)

[6.2 Diagnosis of Current Performance 26](#_Toc260196961)

[6.3 Implications for System Improvement 26](#_Toc260196962)

[7. The Proposed System 27](#_Toc260196963)

[7.1 Sources of Proposals for System Improvement 27](#_Toc260196964)

[7.2 Operational concepts of proposed systems 28](#_Toc260196965)

[7.4 Proposed System Functional Architecture 31](#_Toc260196966)

[7.5 Proposed System Physical and Operational Architecture 40](#_Toc260196967)

[8 Testing and Evaluation 45](#_Toc260196968)

[8.1 Proposed System Metrics 45](#_Toc260196969)

[8.2 Qualification Matrix 46](#_Toc260196970)

[9. Risk Management 46](#_Toc260196971)

[9.1 Risk Identification and Analysis 46](#_Toc260196972)

[9.3 Risk Mitigation Strategy 49](#_Toc260196973)

[10. Technical Program Planning, Implementation, and Control 49](#_Toc260196974)

[10.1 Work Breakdown Structure 49](#_Toc260196975)

[10.2 Project Schedule 50](#_Toc260196976)

[10.3 Budget Overview 50](#_Toc260196977)

[11 Recommendation and Key Issues/Future Directions 51](#_Toc260196978)

[11.1 Analysis of Alternative Approaches 51](#_Toc260196979)

[11.2 Outstanding Issues 52](#_Toc260196980)

[11.3 Future Directions 52](#_Toc260196981)

[12 Discussion of Selected Approach 53](#_Toc260196982)

[13 References 53](#_Toc260196983)

[14 Appendices and Applicable Documents 54](#_Toc260196984)

[1: Project Description 54](#_Toc260196985)

[2: Interview Transcript 55](#_Toc260196986)

[3: Interview Transcript 56](#_Toc260196987)

[4: Interview Transcript 56](#_Toc260196988)

# Part I. Analysis of the Current System

# Introduction and Background- System Context

## **1.1 Unit’s underlying mission**

The goal of this project is to provide a system engineering and workflow outline of the DiscoverArchive system of the Vanderbilt Library. The mission of the Vanderbilt Library is to provide an academic center for the collecting and sharing knowledge and information at Vanderbilt University. This involves maintaining and advancing informational systems and a dedicated staff to serve the Vanderbilt academic community. The DiscoverArchive system fits into this mission as an integral part of the Library’s informational systems. The goal of DiscoverArchive is to create an open access portal and storage system for the uploading and sharing of all research work created by Vanderbilt faculty, staff, and students. A successfully engineered system will promote awareness of the open access movement and increase the likelihood of Vanderbilt embracing such a system.

## 1.2 Unit current status

The DiscoverArchive system is currently in early developmental stages and is managed entirely by Ronee Francis. Ronee is in the process of trying to gain acceptance in the Vanderbilt community for an open access digital information system. She has had success in explaining the concepts to her superiors, but a well-engineered system and workflow outline would greatly improve the chances for acceptance. Additionally, a marketing plan for gaining faculty support of an open access system would also improve the likelihood of acceptance.

## 1.3 Unit priorities

The top priorities for the DiscoverArchive system are to create a system and workflow outline to handle the input of information, to create a system and workflow outline for the storage of information, and to gain acceptance within the university community. The DiscoverArchive system has several advantages working in its favor such as the acceptance of similar open access systems at well-respected universities such as Harvard and MIT as well as a general movement toward the open access of information. Unfortunately, the DiscoverArchive system faces headwinds in the form of the university’s natural resistance to change, the faculty member’s lack of free time, and the resistance of publishers to make their information open access.

## 1.4Unit’s core competencies

The Library has tremendous core competencies in its cataloging capabilities. This is central to the function of a library in being able to successfully and efficiently distribute information to its patrons. The Digital Archive Manager informed our group that once she is able to collect a digital document, she then passes it along to the catalogers easily for appropriate filing and storage. This core competency falls under the operational leadership because the staff is well trained to catalogue the material, and it is their dedicated job. The human capital dedicated to this specific task makes the system’s center of gravity fall under the ‘process-centered’ category because they have the human capital needed to efficiently run the processes. Because the Library is dealing with such a vast amount of information in various mediums, it is essential that they be able to maintain the information in a well-organized fashion. The already process-centered nature of the Library lends itself well to the DiscoverArchive system project in the sense that we as a group can capitalize on the Library’s already effective processes to help move them into a new informational medium (digital open access storage).

## Role of project

This project directly advances the Library’s mission to provide the Vanderbilt academic community with excellent access to information. The open access movements would promote the sharing and storing of digital documents particularly in research fields. This system would be immensely valuable to future academic work and research at Vanderbilt University by cutting down on the dependence of paid for informational systems such as published journals, which is a significant load on the Library’s budget. The project is technologically feasible, especially considering the Library’s core competency in its cataloging ability. While the project is not currently a priority for the Library or the university, the digital archive manager believes firmly that it should be. The basis for this is that more and more information is being created in a digital format, and the Vanderbilt community should try to maintain open access to digital information that is created within the university in order to cut down on access costs to publishers.

# 2. General Description of System Architecture

## 2.1 Operational Concept

The strategic goals of the project are to design a system that will enable Vanderbilt to be a leader in open access capabilities. Open Access is slowly gaining more and more traction but when it takes off, our client wishes to make sure that Vanderbilt is not left behind. Other alternatives might be to wait for the system to develop elsewhere and then copy it, but because of the numerous stakeholders and thus potential roadblocks, waiting for the system to be developed elsewhere is not a viable option given the objective of staying ahead of the curve.

## 2.2 Stakeholder identification and roles

Our project mission is effectively to help Vanderbilt Archives move into a new market segment so the entire project is still in the concept stage and thus all of our stakeholders are present in the concept stage although many will reappear in later stages as well.

Current Phase Stakeholders include:

-Professors

-This stakeholder produces the research commodity around which the system is based. They need to be able to quickly and easily submit the research so that this system does not place a burden on them.

-Publishers

-With the transition to a digital repository, publishers are important stakeholders because they could see a change in their business model. They are concernedabout losing their monopoly power of rights overresearch so it is important to analyze the system with them in mind.

-Vanderbilt IT (Servers etc)

-The system will run on Vanderbilt’s IT system. As a stakeholder they are concerned about both physical and human capital that would be needed to operate the system in the future.

-Vanderbilt Students

-Vanderbilt students represent one of three major customer blocks. They are stakeholders because they access the published research for their own work. Additionally, some students have a stake similar to professors as their senior design/research projects can be archived by the system.

-Other Students

-Other students represent a second large block of consumers, and as such they are concerned with system output. They will need to access the system from external means such as Google.

-General Public (non-current student)

-Even more so than other university students, the general public will need an easy method to interact with the system so that they can truly benefit from the goals of open access.

-Other University Libraries

-Other university libraries will model Vanderbilt’s Discover archive if it is successful and if there is access to the system engineering behind the system.

-Research Subjects

-Paid or unpaid research subjects could gain an understanding of how research publishes findings based on studies to help quell any anxieties. If Discover Archive has good publicity, it could be a valuable tool to these subjects and the researchers that study them.

Future Stage: Open Access Mandate

-Faculty Administration (Faculty Senate)

-The Faculty Senate has control of any policy change from the producer (Professor) side. They care about the system from an ideological standpoint regarding what should be done in higher education and are likely to be heavily influenced by peer institutions.

-University Administration (Provost)

-The Provost has an important voice in Open Access policy as well. He has many of the same concerns as the faculty senate but also must consider budgetary and other administrative concerns.

-Educational Policy Makers

-The Discover Archive system may be impacted by education policy decisions, and these decisions regarding Open Access may be influenced by the forerunners of the Discover Archive movement.

-Open Access Forums (non-university)

-Open Access forums are generally nonprofit ventures that seek to proliferate knowledge. While they have no direct knowledge, if Discover Archive takes them into consideration, it might form a valuable relationship that could provide quality feedback in the future.

## 2.3 External Systems Diagram

The external systems diagram shows the system interacts with external but related systems. The links represent the means in which the systems interface. The End User, which might be a college student trying to read published research accesses the DiscoverArchive repository through its servers. Faculty use a special authorized link to submit the material to the system, which at all times maintains a secure network connection to the Vanderbilt Server System. Librarians monitor the DA system through dedicated system computers.

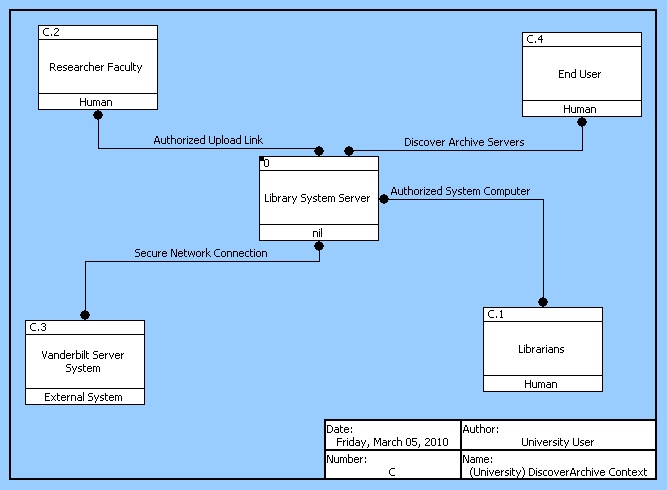


Figure 1: External Systems Diagram

When Vanderbilt research faculty generate a paper, they currently either input it directly into Discover Archive, or give it to a publisher as represented by the two links from research faculty. Publishers then present the papers in various physical journals but can send a digital copy to the Discover Archive by communicating with the staff. The DA system is housed on Vanderbilt’s servers, and the End User (student, curious citizen, etc.) then connects via the internet to the DA system.

## 2.4Identification of Comparable Systems for Benchmarking

The spread of popularity of the internet as a mechanism for information sharing led to the growth of the concept of open sharing of research publications in the 1990’s. Since then there have been multiple institutions and societies who have promoted this movement and looked to create central databases with worldwide access. In 2001 over 30,000 scholars signed a petition calling for a formal organized movement towards open access, and since that point many organizations have officially recognized some degree of participation in the movement.

### 2.4.1 Benchmarking partner criteria

Of these organizations, we feel that it would be ideal to focus on similar universities with strong research initiatives in order to appropriately benchmark the progress of Vanderbilt’s Open Access initiative. Potential Benchmarks must be a Research I University, have an established policy of the Open Access with a corresponding digital repository system, and still maintain relationships with the leading print journals. The first criterion will generally (but not always) ensure that potential benchmarks have similar input in terms of quality and quantity of research, and that the universities have similar stakeholders, requirements, and constraints. The second criterion will ensure that the benchmark is actually a benchmark, that is to say that they are further developed than Vanderbilt system. By having an established Open Access policy and repository, the benchmarks will illuminate some design elements that the Discover Archive system should incorporate as well as some elements that the Discover Archive system will need to improve upon. The final criterion is very important because the fiscal and legal relationship with established print journals dictates the success of the Open Access database. Print journals still play a large role in the process even if the papers are published to the Open Access database. The journals lend credibility to research by publishing it so they are still very important to professors, and we would like to learn from and benchmark ourselves on institutions that have a working knowledge of this relationship.

### 2.4.2 Benchmarking partner identification

Harvard University hasdefined and enacted a system by which we can benchmark our development. Harvardimplemented their program in 2008 and thus have approximately two years of experience. Duke has an office of scholarly communications that has been at the cutting edge of open access policy so they would be an ideal benchmark for the ideological, political, and legal aspects of the system.

### 2.4.3 Benchmarking partner agreement

Duke has agreed to help us with our project. Currently, their system has been proposed but has not been voted on by their faculty. Much like DiscoverArchive, their repository currently holds primarily theses and dissertations so their system is inspirational as well.

Duke does not have plans for a specific addendum, because most journals already allow self-archiving in an institutional repository. When the journal does not permit this, generally the policy/license is waved, so there is only a small group for whom the addendum would be valuable. Duke staff believes that publishers tend to reject addenda out of hand, so they advocate for a direct request for needed rights which the publisher would then write into the regular contract. As for the management of the repository system, it is envisioned to be based around a small group of librarians that will receive citations and hopefully the PDF. In any case, the aim is to make it easy for faculty, so that they only need to say OK to a proposal that a given set of articles be deposited.

This system is applicable to DiscoverArchive will be valuable in consideration of DiscoverArchive improvements from a system engineering perspective.

# 3. Requirements Analysis

## 3.1 Operational Scenarios

3.1.1 System acquires research document

3.1.2 System acquires alternative media type

3.1.3 System server fails

3.1.4 System manager encounters rights dispute

## 3.2 Operational Scenarios in Full

3.2.1  Systemacquires research document

The primary operational scenario for the DiscoverArchive system is acquire published research documents of Vanderbilt faculty. Because there is no existing mandate for submission to the archive, the documents must be located and manually added to the archive. In some cases the documents are submitted by the authors, but usually the archive managers must search them out.

     - Digital archive manager obtains viable research document

     - Manager sends addendum to faculty for approval

     - Faculty signs and resubmits addendum

     - Manager submits addendum to publisher

     - Manager checks publisher database for open access status

3.2.1.1  Publisher allows open access publication

In this best case scenario, a publisher allows immediate open access publication of a post-print document.

            - Post-print document is sent to catalogers

            - Catalogers complete uploading of document to the system

            - Manager notifies faculty member of open access publication

3.2.1.2  Publisher does not allow open access

In some cases, a publisher will restrict open access publication.  In these scenarios, the system can try to negotiate some authorship rights.

           - Post-print document is maintained within system, but not uploaded

            - Manager notifies publisher of open access movement

3.2.1.3  Publisher requires hold period before open access publication

There are also some scenarios where a publisher requires a certain length of time before a document can be made open access.

           - Manager checks publisher database for specific requirements

            - Document is maintained within database for a length of time

            - After time period, document is sent to catalogers and uploaded

            - Faculty member is notified of open access publication

3.2.2 System Server fails

3.2.2.1 Internal Failure (DA manager works to resolve issue)

3.2.2.2 External Failure (DA manager contacts ITS)

3.2.3  Archive acquires alternative media type

There may be some instances where a faculty member creates or submits research that is not in the form of a PDF or a Word document.  These cases may require specific attention from the DiscoverArchive system in terms of handling, cataloging, and publisher querying.

3.2.3.1 Faculty member submits alternative media type needing attention (film, audio, etc.)

            3.2.3.1.1 System manager generates media specific addendum for faculty

            3.2.3.1.2 Manager sends addendum to faculty for approval

            3.2.3.2 Facultysubmits alternative media with appropriate digital rights

 At this point the system will proceed as above starting with one of 3.2.1.1, 3.2.1.2, or 3.2.3.3 depending on the specific requirements of the individual publisher.

3.2.4 System encounters digital rights dispute (handled by DA manager)

## 3.3 Input/Output (I/O) Trace

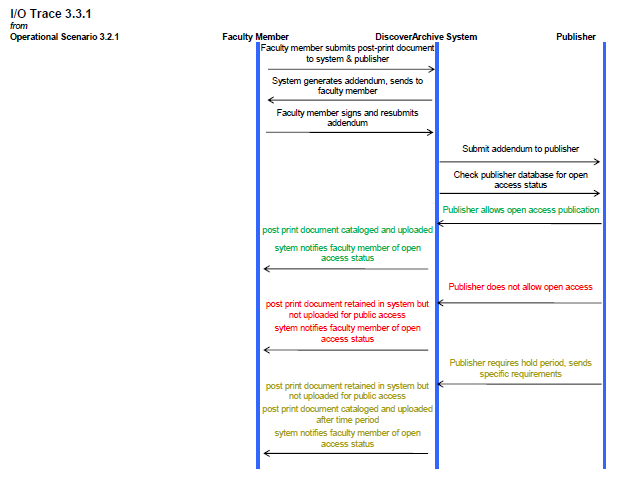


Figure 2: I/O Trace for Operational Scenario 3.2.1 shows the inputs and outputs to and from the system when it receives a document. The green inputs and outputs represent a publisher that allows open access. The red inputs and outputs represent a publisher that does not allow open access, and the yellow inputs and outputs represent a publisher that conditionally allows open access.

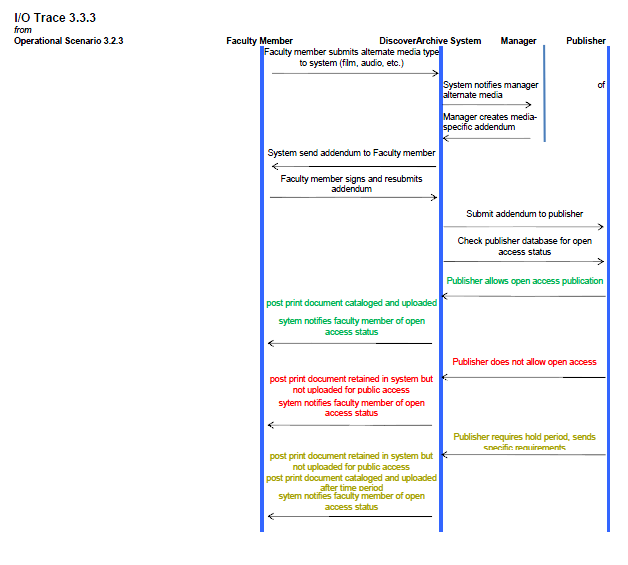


Figure 3: Input/Output Trace for Operational Scenario 3.2.3 shows the inputs and outputs to and from the system when it receives alternate media formats. The color coding is the same as in the above figure.

## 3.4 Complete Requirements Hierarchy

3.0 The system shall work as decided by the system manager.

3.1 The System shall archive documents

3.1.1 The system shall receive documents for open access publishing.

3.1.1.1 The system shall receive documents when submitted by professors.

3.1.1.2The system shall receive documents from publishers

3.1.1.3The system shall grant access to the librarians so that they have access to the database

3.1.1.3.1 The system manager shall edit received documents to be suitable for cataloging.

3.1.1.3.2 Catalogers shall make edits and cataloging descriptions before publishing documents to open access.

3.1.2The system shall accept alternative media source submissions from faculty.

3.1.2.1The system manager shall appropriately edit the alternative media source.

3.1.2.2The system manager shall post alternative media to open access.

3.2The system shall perform routine self-maintenance.

3.2.1The system shall maintain a functional DiscoverArchive website.

3.2.1.1The system manager should be able to process 30 documents per month.

3.2.1.2The system website should be able to accommodate traffic of 5000 views per month.

3.2.2 The system shall store all open access files in the system server.

## 3.5 Technical Performance Metrics and Target Levels

Frequency: TPMs in the DiscoverArchive system measure the frequency with which a requirement is performed. One of the objectives of the system is to adapt to changing workloads, so having TPMs that ensure the system is updating for possible alternate spellings of professor names, or for updating the list of non-compliant professors is essential to managing the number of cases that do not conform to the intended system. These TPMs are measured in units of time such as days or weeks.

Data size: The other important TPM is data size. In the event of a system failure, emergency holding pens will be able to accept incoming submissions but not at the rate of the full system. Thus this requirement is measured by data capacity. Such TPM’s are in units of Giga Bytes.

## 3.6 Requirements Priority Tree

The requirements priority tree shows the requirements at each level of detail. Each requirement is weighted by relative importance, so it follows that all of the requirements at any level of priority add up to one. Within a level, the requirements with the greatest priority have the highest numeric priority value, which conveys their relative importance.

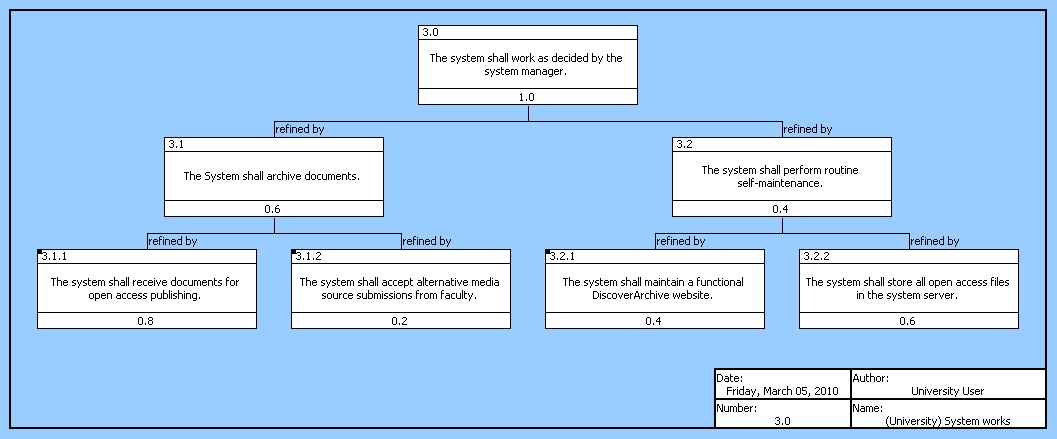


Figure 4: Requirements at the 3.0 level.

Figure 4 shows the overall system requirement (3.0) and then the first and second levels of requirements that refine the first level. It helps the visualize the priorities of the requirements.

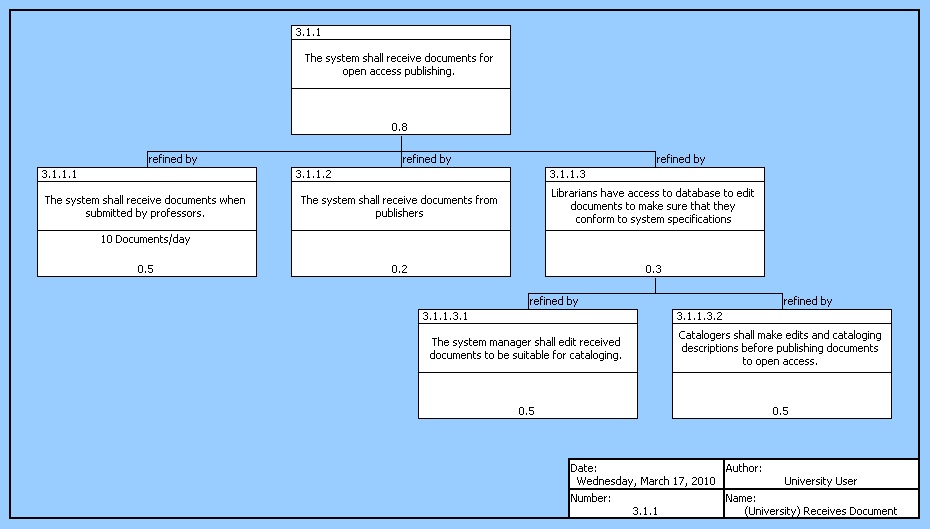


Figure 5: Requirements hierarchy at the 3.1.1 level.

Figure 5 shows the fourth and fifth level requirements that refine requirement 3.1.1., which is one of the most instrumental requirements (as evident by the fact that its weighting is double that of any other third order requirement. This diagram shows the requirements that must be taken into consideration since they refine the 3.1.1 requirement.

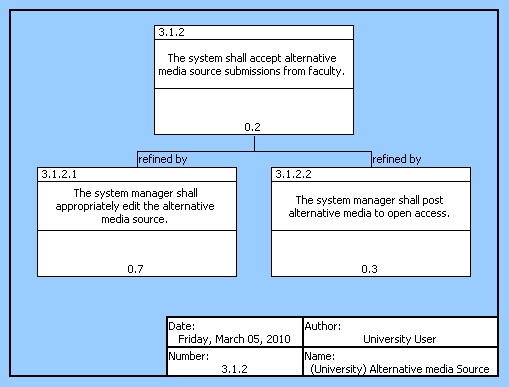


Figure 6: Requirements Hierarchy at the 3.1.2 Level.

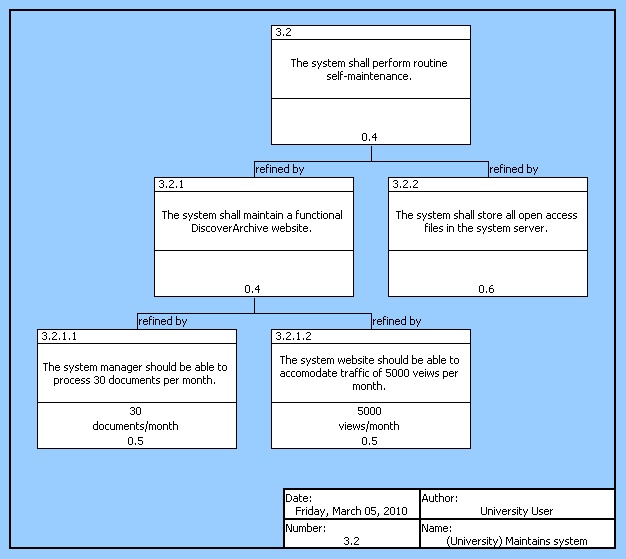


Figure 7: Requirements Hierarchy at the 3.2 level

Figures 6 and 7 also show requirements hierarchies which visualize the relative priorities of system requirements. Technical performance measurements quantify performance for the system.

# 4 Functional Architecture

## 4.1 System Functional Diagram

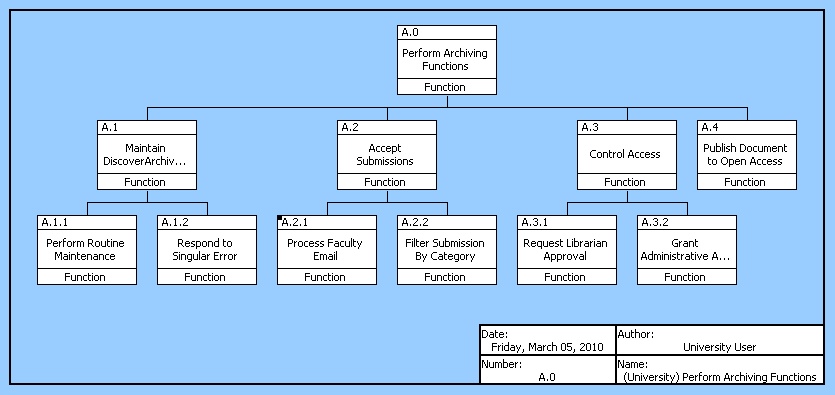
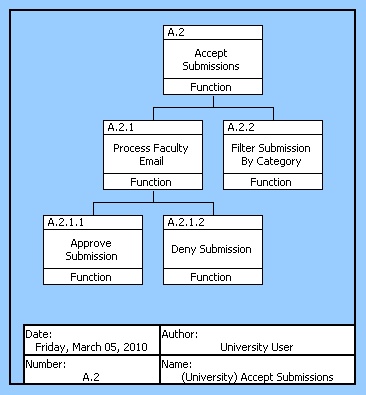
****

Figure 8: Functional Hierarchy at A.0



Figures 8 and 9 show functional hierarchies for the system. Higher level functions are decomposed by functions contained in them that help the high level function execute its aim. A.2 Functional Hierarchy is highlighted because of the importance of accepting submissions for the system. It is decomposed by secondary and tertiary functions that aid in A.2 complete its task of accepting submissions.

Figure 9 Functional Hierarchy at A.2

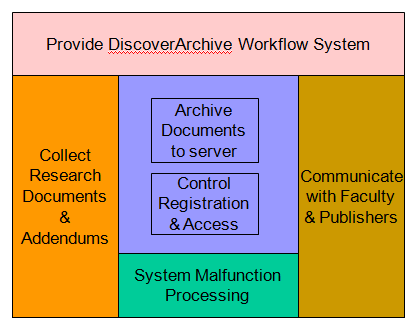


Figure 10: Hatley-Pirbhai diagram

Figure 10 helps explain the functions in the system as well as the inputs and outputs of the system.

## 4.2 Functional Flow Block Diagram

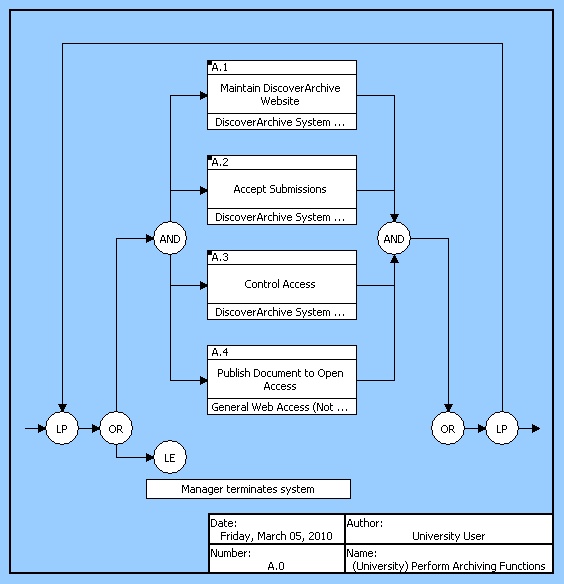


Figure 11: Functional Flow Block Diagram at the A.0 level.

Figure 11 shows the functional flow logic of the system in the broadest sense. It shows the logic of the four main functions of the system always working in parallel. They exist in a loop because they are continually being performed unless the loop exit condition is met, in which the system manager terminates the system.

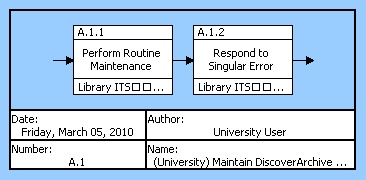


Figure 12: Functional Flow Block Diagram of function A.1 shows the two functions that exist within the function of maintaining the DiscoverArchive website.

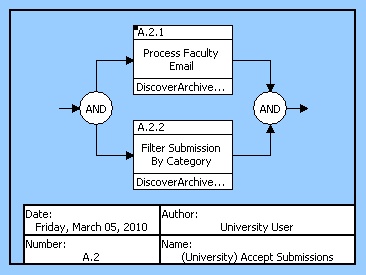


Figure 13: FFBD of function A.2 showing what occurs within the accepting submissions function.

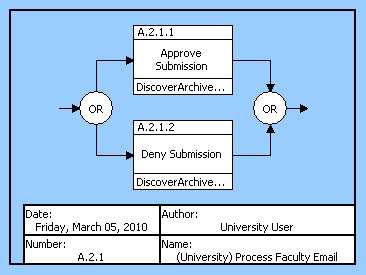


Figure 14: FFBD of function A.2.1

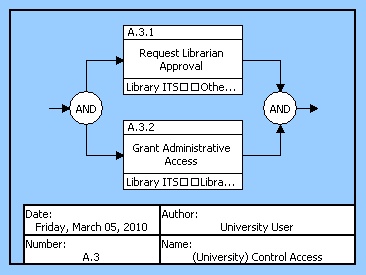


Figure 15: FFBD of function A.3

Figures 13 and 14 show the functional flow within the A.2 function. Figure 13 displays the parallel execution of A.2.1 and A.2.2 while A.2.1.1 and A.2.1.2 functions in figure 14 operate through an OR construct because only one of the two functions executes.

## 4.3 Enhanced Functional Flow Block Diagram

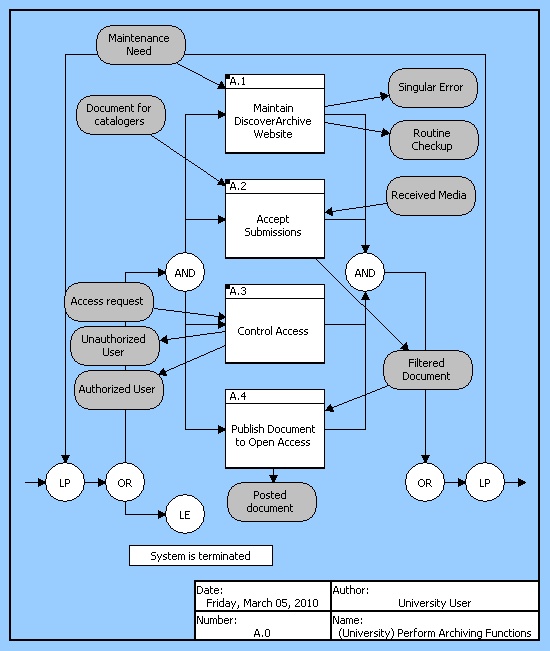


Figure 16: Extended Functional Flow Block Diagram at the A.0 level

Figure 16 shows the A.0 function decomposed into its first level functions and their respective inputs and outputs.

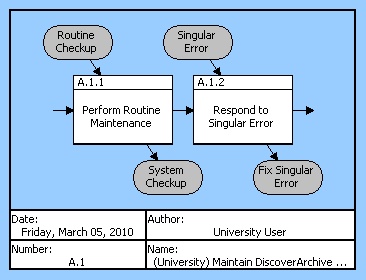


Figure 17: EFFBD at A.1 Level displaying the various inputs and outputs that are associated with their respective functions.

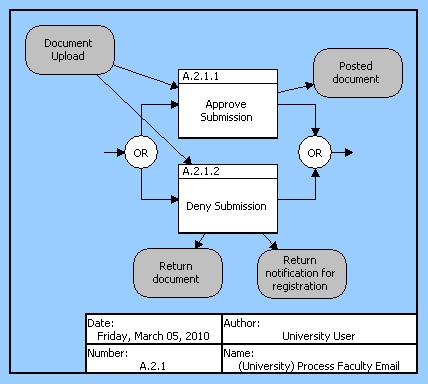


Figure 18: EFFBD at A.2.1 Level shows that when a document is uploaded to the system manager, it must be either approved or denied. If approved, the document can be posted, but if the document is rejected, it will be returned along with a return notification.

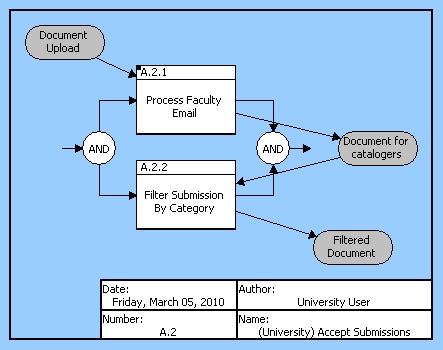


Figure 19: EFFBD at A.2 Level shows that when a document is uploaded, it is processed by the system manager and then becomes a document for the catalogers. Catalogers then perform the function of filtering the document resulting in a final filtered document.

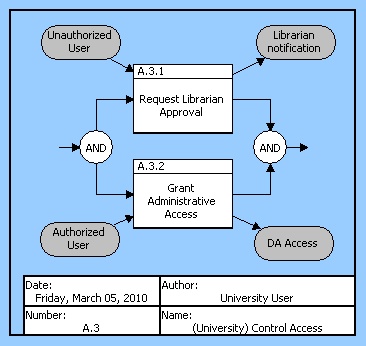


Figure 20: EFFBD at A.2 Level explains how users will be either granted or denied access to the system.

# 5 Physical and Operational Architecture

## 5.1 General Architecture

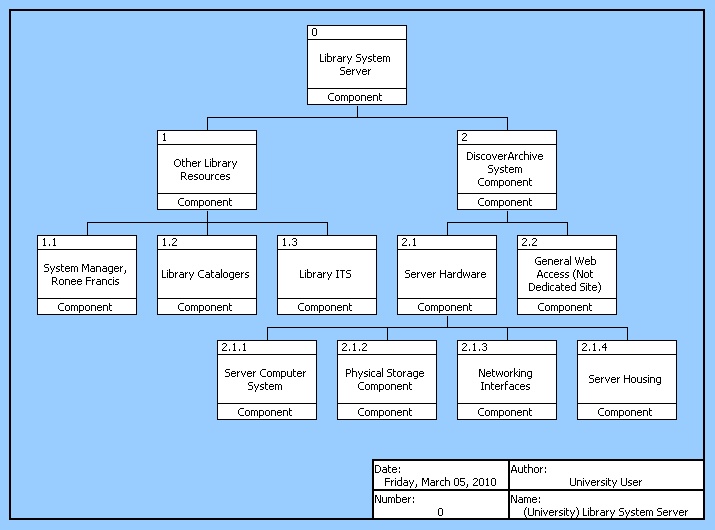


Figure 21: General Architecture

Figure 21 shows the architecture of all physical components of the system. This visualization allows for easy understand of the order in which physical components need to be build or acquired for the system to work, and makes sure that all physical components are accounted for.

## 5.2 Operational Architecture

|  |  |  |
| --- | --- | --- |
| Requirements | Functions | Components |
|  |  |  |
| System works | Perform Archiving Functions Accept Submissions Control Access Publish Document to Open Access | Library System Server Other Library Resources DiscoverArchive System Component |
| Archive Documents | Perform Archiving Functions Accept Submissions Process Faculty Email Approve Submission Deny Submision Filter Submission by Category | Library System Server System Manager, Ronee Francis Library Catalogers Server Hardware Server Computer System Physical Storage Component Server Housing |
| Receives Document | Accept Submissions | System Manager, Ronee Francis Server Hardware |
| Research from professors | Accept Submissions | System Manager, Ronee Francis |
| Research from publishers | Accept Submissions | System Manager, Ronee Francis |
| Librarians edit received document | Process Faculty Email Filter Submission by Category Request Librarian Approval | Library Catalogers System Manager, Ronee Francis |
| Manager edits | Accept Submissions Process Faculty Email Filter Submission by Category | System Manager, Ronee Francis |
| Cataloger edits | Accept Submissions Process Faculty Email filter Submission by Category | Library Catalogers |
| Alternative media source | Accept Submissions Process Faculty Email Filter Submission by Category Publish Document to Open Access | System Manager, Ronee Francis Server Computer System |
| Media Handling | Filter Submission by Category Publish Document to Open Access | System Manager, Ronee Francis |
| Post alternative media | Publish Document to Open Access | Library Catalogers System Manager, Ronee Francis DiscoverArchive System Component |
| Maintains system | Maintain DiscoverArchive Website Perform Routine Maintenance Respond to Singular Error Control Access Request Librarian Approval Grant Administrative Access | System Manager, Ronee Francis Library ITS Server Hardware Server Computer System |
| Website maintenance | Maintain DiscoverArchive Website Perform Routine Maintenance | Library ITS System Manager, Ronee Francis Server Hardware Server Computer System General Web Access |
| Document uploads | Publish Document to Open Access | General Web Access Library Catalogers System Manager, Ronee Francis Networking Interfaces |
| Open access | Publish Document to Open Access | Networking Interfaces System Manager, Ronee Francis Server Hardware |
| Data storage | Maintain DiscoverArchive Website | Server Hardware DiscoverArchive System Component |

Figure 22

Figure 22 shows the tracing of all requirements, functions and components in the system. This traceability chart ensures that all requirements are met by functions that are mapped to components which facilitate the completion of the requirements. For example, the requirement, “Librarians edit received document” is completed by the functions, “Process Faculty Email”, “Filter Submission by Category”, and “Request Librarian Approval”. In turn, these functions are completed by the library catalogers and the system manager. Such a table helps show how functions are completed and can be used to show any duplications or inefficiencies.

# 6 Current System Performance

## 6.1 Technical Performance Measures

The existing system is almost completely human and is not governed by any performance requirements as it stands, so current performance cannot be judged against any technical performance measures as of yet. If there were an existing mandate for University faculty to upload documents to the Discover Archive then the progress and performance of the human components could be judged on the volume of documents that they are able to upload in a given time period. However, documents are uploaded and edited as Ms. Francis come across them, and she generally works at her own pace in a continuous effort to increase document volume in the database.

This analysis leads us to the conclusion that the automated system to be implemented will be measured by the number of documents per day, for example, that it is able to procure and upload to the server. It will also be judged on its ability to enforce registration and submission requirements. The current system simply cannot be judged based on these criteria because there are no set performance targets.

## 6.2 Diagnosis of Current Performance

The current system functions acceptably as a low-volume process for document procurement and upload to the DiscoverArchive server. However, the true purpose of the archive demands a much more streamlined workflow and automation process to remove a majority of the human element from the system. It is this idealistic functionality against which we compare the existing system, for the existing system itself has no set requirements. It is anticipated that an improved system will encourage a University-wide mandate for all Vanderbilt research to be published to the archive, following in the footsteps of other top research universities such as Duke and Harvard. When this happens, the document volume will increase exponentially and will be far too much for the current human based system to handle because of critical flaws in the workflow.

The current system relies on one human component to look for potential research documents for the archive. This method does not provide anywhere near the system capability necessary to process the large number of documents being output by the University as a whole on a daily basis, and presents the first major flaw in the current system. A single person is simply not able to monitor all of the research outlets and publications to determine what should be uploaded to the archive. Application of the “five why’s” technique leads to the root cause of this particular system deficiency:

The system is not capable of handling document volume

1. *Why?* The massive volume of research documents cannot be processed.
2. *Why?* The system relies on a single human element to perform document acquisition.
3. *Why?* There is no computerized system to automate the process.
4. *Why?* There is no mandate for DiscoverArchive.
5. *Why?* The Open Access initiative is just now developing in research institutions.

This analysis shows two important root causes for this issue, that there is not yet a mandate for the DiscoverArchive which is a result of the fact that the concept is still in its developmental stages for academic institutions. The latter issue is being rendered irrelevant as peer research universities implement successful systems, and as Vanderbilt recognizes the success of this system at other schools it will move towards the mandate that will initiate the creation of a fully capable open access system.

The second major flaw in the existing system that must be addressed is the need for a human element to monitor the publishing rights of a given document to determine whether or not the document can be uploaded to the server. The flaw in this process arises from the need for multiple communications with various entities to secure publishing rights, which is a time consuming and difficult task for a human element to perform on a large scale. An ideal system would replace the human element with an automated process that could automatically handle all communications as documents are submitted to the archive.

## 6.3 Implications for System Improvement

The implications of the existing system flaws discussed above point towards the definite need for an automated system to efficiently process the large volume of documents that is expected with a university mandated move to open access. While the existing system can decently handle the current number of documents, it is no way prepared for a full transition to open access. In order to be ready for this, DiscoverArchive must be re-designed to perform the majority of tasks in an automated manner with a dedicated server and website that is capable of carrying out all of the functions currently performed by human elements.

The primary aim of the automated system will be to process and upload the massive number of documents that will be handled by DiscoverArchive. Replacing the human elements will be a server with software to collect submissions and monitor journals for documents to be posted to the archive, thereby eliminating the first major flaw in the existing system. An automated system will have no relevant limit to the number of documents it can process, meaning that it will be adaptable to any anticipated future expansion of the archive.

The second major flaw in the existing system would again be overcome by the implementation of automated server functions. The acquisition of publishing rights would be much more efficiently accomplished by an automated process that could handle a larger volume of documents and exponentially more communication requirements. An ideal solution would automatically send an addendum with the document to the publisher to secure rights, and once granted the system would upload all relevant supporting documents including the addendum and the publisher’s PDF with the original document to the DiscoverArchive server. The automated system would circumvent the inevitable bottleneck in document flow that would occur in the current implementation of the system because the majority of the process would be automated. The human element would only be required to interact with the process under certain extenuating circumstances that required special attention.

We currently have no technical performance standards to compare this system to, other than general statements about the overall functionality of the system as compared to solutions implemented at other universities. Both Duke and Harvard have implemented systems that incorporate automated processes for handling document submission and rights acquisition. The functionality of these systems is clearly far beyond the capabilities of Vanderbilt’s existing DiscoverArchive, but we are confident that after addressing the archive’s inherent flaws and implementing a similar automated process the Vanderbilt DiscoverArchive will be ready to handle a full university transition to open access.

# 7. The Proposed System

## 7.1 Sources of Proposals for System Improvement

The system as it currently exists functions well enough to handle the research that is acquired by the system manager. However, the true goal of the DiscoverArchive is to enable open access for all research generated at Vanderbilt. Reaching this goal will certainly require the passing of a university mandate, but if a mandate were passed, the existing system would be overwhelmed with incoming information. For this reason, the remainder of this report will focus on possible changes that can be made to the existing system to improve its functionality.

One method for determining positive changes for the system is using ESIA analysis, which stands for Eliminate, Simplify, Integrate, and Automate. In such analysis, we believe that certain aspects regarding the reformatting and inspection aspects surrounding the gathering of research. This can potentially be achieved using an upload website portal where administrators would submit their research to the system manager. However, the website could have restrictions such that only research that was correctly formatted and inspected could be uploaded. We also believe that some of the flows and processes can be simplified in a modified system. Again building off the idea of a website upload portal, if research was ensured to be submitted in a proper format, it could be transferred directly to catalogers without needing to be looked at by the system manager. The long-term success of DiscoverArchive would benefit greatly from integrating university researchers with the system. This will require a process for engaging university researchers in the open access movement and also creating a means for them to register for access to the upload website portal. Finally, we believe that this system could benefit dramatically from automation. Some of the actions required for getting research published to open access is tedious and time consuming and can likely be accomplished through automation. A revamped website will assist in this, but we also believe that some custom designed software scripts could allow the system manager to retrieve, format, and publish alternative media types (specifically from iTunesU) extremely efficiently. Taking this analysis into account, we will proceed with a proposed system that has incremental change, as well as a system involving progressive change.

## 7.2 Operational concepts of proposed systems

The main premise behind our proposed system changes is to eliminate inefficiencies in the current system by automating some of the tasks performed by the system manager, Ronee Francis. By eliminating these routine tasks from the manager’s responsibilities, more time can be used by the manager for promoting the DiscoverArchive system on campus.

*Incremental system change*

In the incremental system change, the most basic of the manager’s responsibilities will be automated. Additionally, the overall system for handling and processing research documents will be limited. For example, there will be a basic website that can be accessed by Vanderbilt researchers where files can be uploaded. However, this website will direct files to the manager for further sorting and publishing to DiscoverArchive. Other simple tasks such as creating metadata templates, creating collections, and assigning catalogers to submitted research can potentially be automated in this system. These changes will take some load off of the system manager.

*Progressive system change*

Our proposal for the progressive system change is to essentially automate all of the system manager’s tasks that can be automated. This would entail a complete overhaul of the DiscoverArchive website to include in depth upload portals for Vanderbilt researchers. Within these portals, submitters will include details about their publications to help create metadata templates and improve cataloging time. The DiscoverArchive website will also include training information for researchers, collection administrators, and undergraduate and graduate students who work on the DiscoverArchive system. In conjunction with these automations, the progressive system change includes a passed university-wide mandate for researchers to submit research to the DiscoverArchive system. The system will then automatically track registration and send notifications to those who have not registered but should. Submitted research will also get automatically cross-referenced with databases containing policies for different publishers. This will remove the responsibility of checking policies from the system manager and automatically flag articles that can’t be posted immediately. In conjunction with the university mandate, when research is submitted that can’t be directly posted to DiscoverArchive due to policy constraints, a contract addendum will be generated to be sent to the publisher in an attempt to regain open access publication rights. Finally, alternative media types, specifically that from iTunesU, will be downloaded handled, and posted automatically using software implementation. The system manager has expressed that downloading, formatting, and posting this data is extremely time consuming, but if software scripts were written to automatically do these steps at prescribed times, it could greatly reduce some burden from the manage**r.**7.3 Proposed System Requirements

*Incremental System*

|  |  |
| --- | --- |
| 3.1 | Archive Documents |
| **3.1.1** | **Receives Document** |
| 3.1.1.1 | Research from Professor |
| 3.1.1.1.1 | *Check* Publisher Regulations |
| 3.1.1.1.2 | *Thank* Faculty |
| 3.1.1.2 | Research from other source |
| 3.1.1.3 | Post-print from publisher |
| 3.1.1.4 | Librarians edit existing system documents |
| **3.1.2** | **Publisher Interaction** |
| 3.1.2.1 | Receive Publisher PDF |
| 3.1.2.2 | *Check* publisher rules |
| 3.1.2.2.1 | Rule decision |
| 3.1.2.2.2 | Publisher notification |
| 3.1.2.3 | *Notify* author of open access |
| 3.1.2.3.1 | Offer opt-out |
| 3.1.2.3.2 | Author feedback |
| **3.1.3** | **Alternative Media Source** |
| 3.1.3.1 | Media Handling |
| 3.1.3.2 | Post Alternative Media |
| **3.1.4** | **Server Malfunction** |
| 3.1.4.1 | *Notify* User |
| 3.1.4.2 | Emergency Holding Pen |

The improvements to the incremental system hinge around automating some of the tasks currently performed by the human components of the system. Any of the simple requirements (see figure 23) that include simple tasks like ‘Check’, ‘Notify’, or ‘Thank’ are additional requirements to this semi-automated system. For example there was no previous requirement to thank the faculty because in the interaction with the archivist, emails would have been exchanged. Here, where there might be less human interaction, the system must thus confirm receipt as to not leave the faculty member guessing whether or not the system performed its job.

Figure 23

*Progressive System*

The new system is designed from the ground up to be fully automatic. Thus the requirements include provisions for the system to archive just as the previous system and incremental system have done, but also to be sustainable from a training and management perspective (see figure 24). Even within the familiar requirements of archiving, Technical Performance Measurements have been set (see figure 25) so as to allow DiscoverArchive managers monitor how the system is performing and make any changes that become apparent if TPM’s aren’t being met. Figure 25 also shows the refining requirements needed to archive documents which is the most important requirement. Other requirements not pictured at the 3.1.X level include archiving ITunesU content, hosting the website, and digitizing submissions. The changes to the overall system also include a third main component beyond, automation and sustainable management, (see figure 24) which requires it to be Open Access Mandate compliant. Although this is not yet likely, this progressive system is designed to be functional farther into the future than any of the other systems.

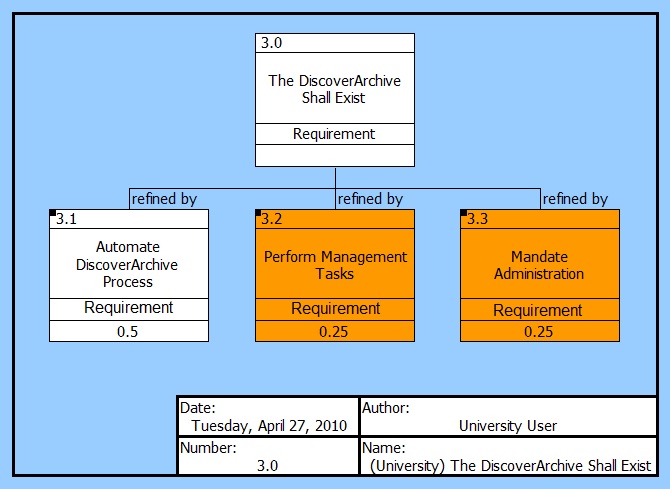


Figure 24- This figure shows two entirely new branches of requirements, highlighted in orange. They represent new requirements of the progressive system which automates a majority of the DiscoverArchive process and thus is required to perform management and administrative tasks as well. However, automating the archiving is still the most importantly weighted requirement.

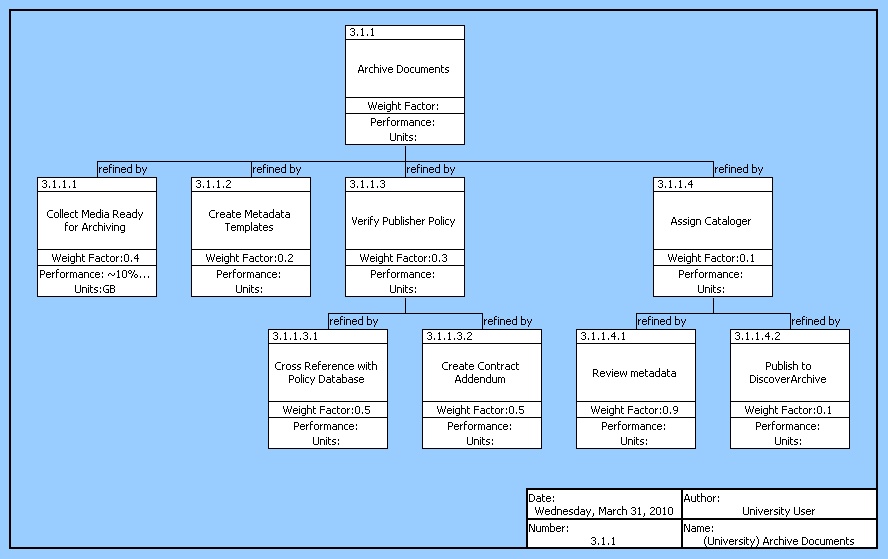
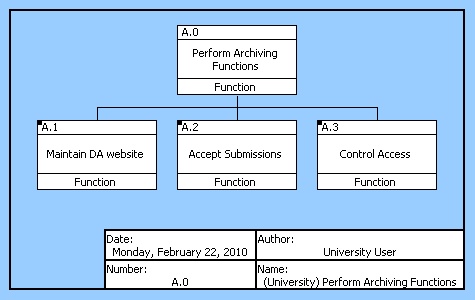


Figure 25: Progressive redesign of the archiving process including TPM

7.4 Proposed System Functional Architecture

*Incremental system*

The incremental system contains all the same components (not pictured), and some functions (see below) have been updated so as to include automation. Figure 26 shows the three main functions.

Figure 26

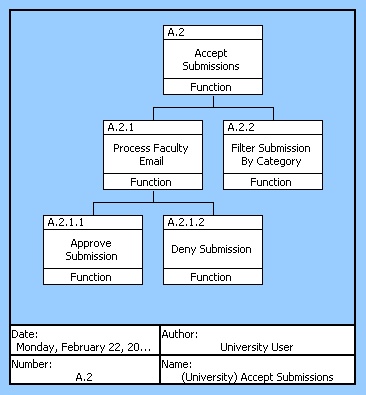


Figure 26

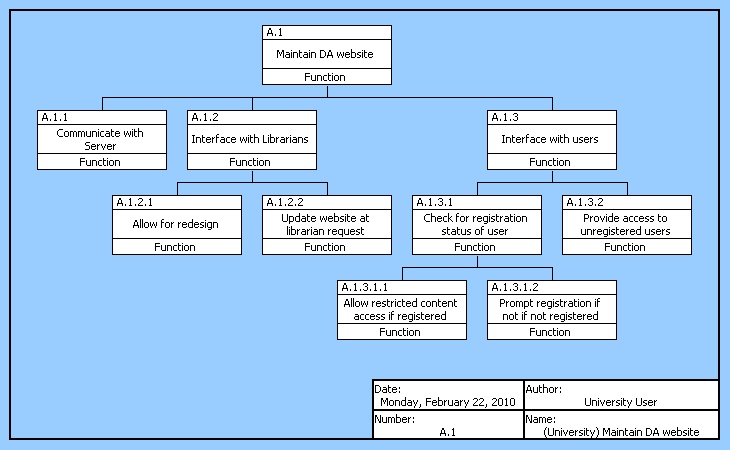


Figure 27

Figures 26 and 27 detail some of the functions that automate parts of the system. Functions in Figure 26 automatically accept or deny a submission which allows librarians to quickly upload the approved documents, and look for ways to make the denied submissions permissible. Figure 27 introduces a user registrations system so that registered users can automatically submit subsequent research since they already understand the process which will take some workload off of the librarians.

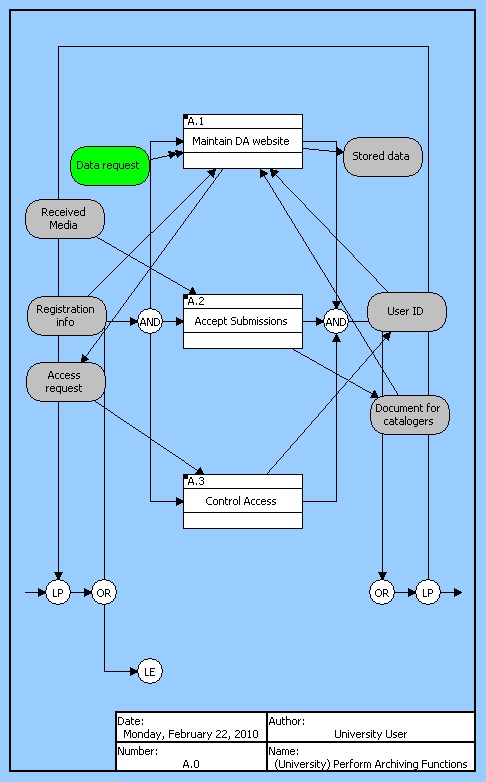


Figure 28

Figure 28 shows the inputs outputs and triggers that control these functions. The entire system runs on a repeating loop although there is a loop exit should the library ever decide to end the system. The three first level systems operate simultaneously, and there is a circular flow of input and output between the website and website access functions as they work together to help the system disseminate information. Also, as media is received it passes through the receiving function where it reaches the catalogers who input it into the website where it enters the website access circular flow.

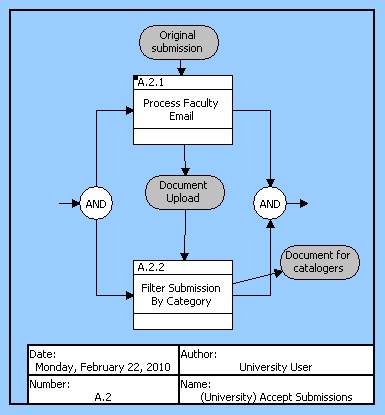


Figure 29 shows the internal logic of A2 function from figure 28

New submissions from email are simultaneously internalized into the system and filtered. Then the document outputs to the catalogers who take the document back to the A1 function that maintains the website.

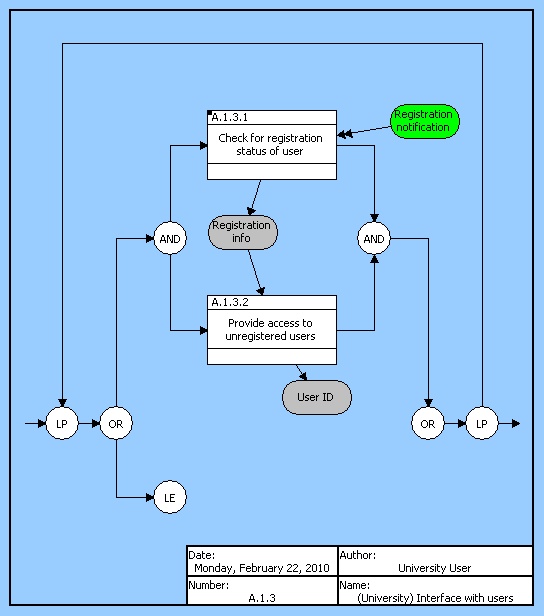


Figure 30 Explanation of the new registration process

Figure 30 explains the new registration process that is meant to lighten librarian workload by granting a ‘fast pass’ to returning users who know how to correctly submit to the DA system.

*Progressive System*

The functions in the progressive system are designed to fully automate the system, make it self-maintaining, and make it sustainable far into the future. Figure 31 shows the three main functions of the system. Automating the DiscoverArchive process is a priority, but it need supporting functions to help manage the non automated tasks including training employees on the system, as well as a function to help the system prepare for an open access mandate. Figure 32 decomposes the automation process into four functions that automate different processes. Figure 33 provides a good example of how automation works as many functions work to archive the material but only one involves the cataloger. Figure 34 shows the inclusion and automation of functions that will facilitate an open access mandate should it come to pass.

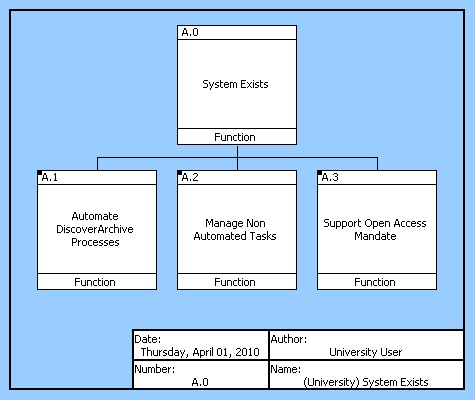


Figure 31

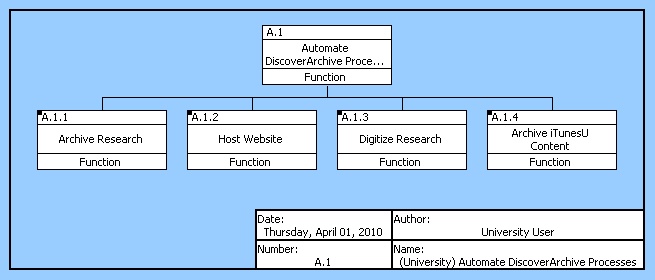


Figure 32

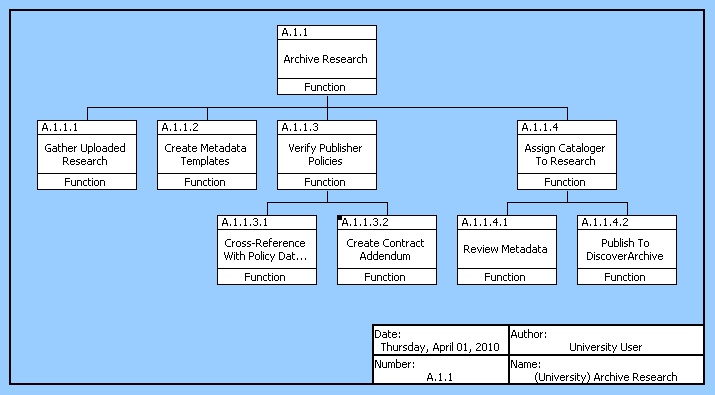


Figure 33

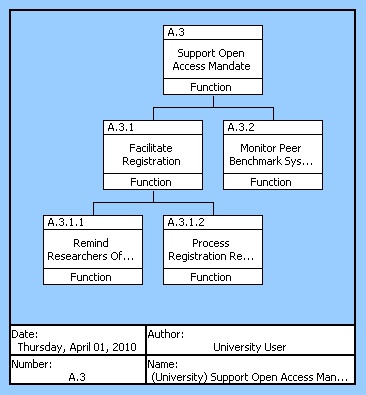


Figure 34

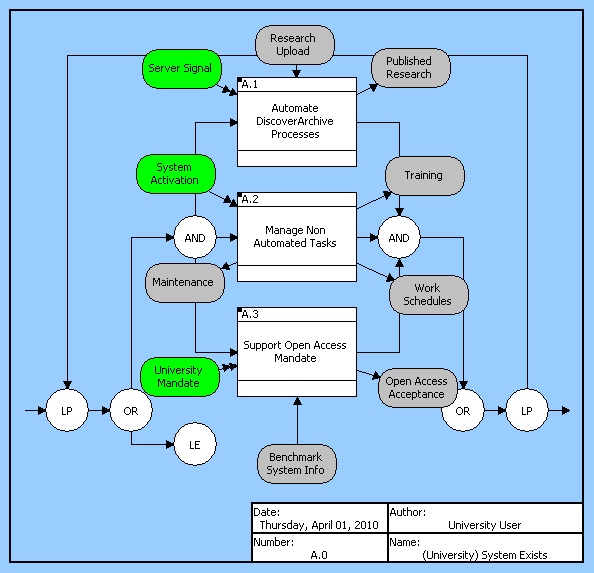


Figure 35

Figure 35 shows the input outputs and triggers that control the functions. For example, the open access function is included in the and loop but does not operate unless the University open access Mandate triggers it to do so. The system outputs published research, but outputs behind the scenes outputs such as work schedules and training for the human components that are still needed in some parts of the system.

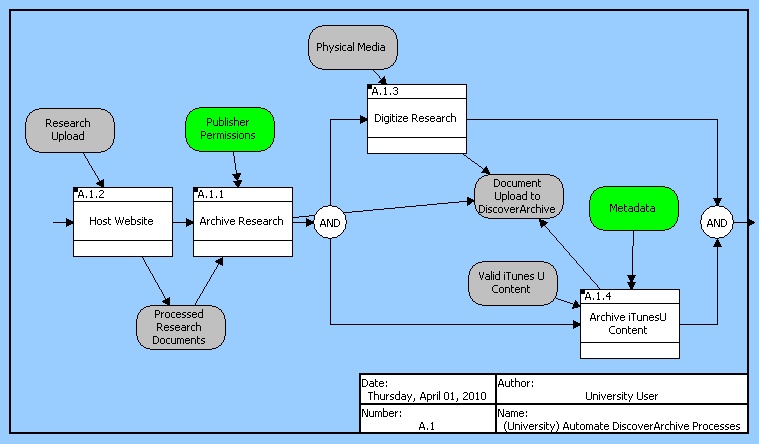


Figure 35

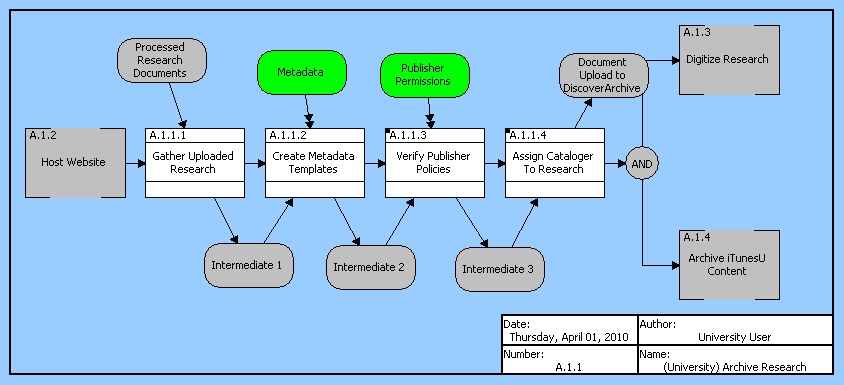


Figure 36

Figure 36 shows the functions contained in the Archive Research function of Figure 35. As a whole, figure 35 shows an automated flow of the document through the archiving processes. Given the publisher permission trigger, research can be archived for preservation and also uploaded for dissemination. During the archiving process, as figure 36 indicates, the functions automatically gather research, create templates if metadata is available, verify publisher policies, and then produce a document to be uploaded and archived.

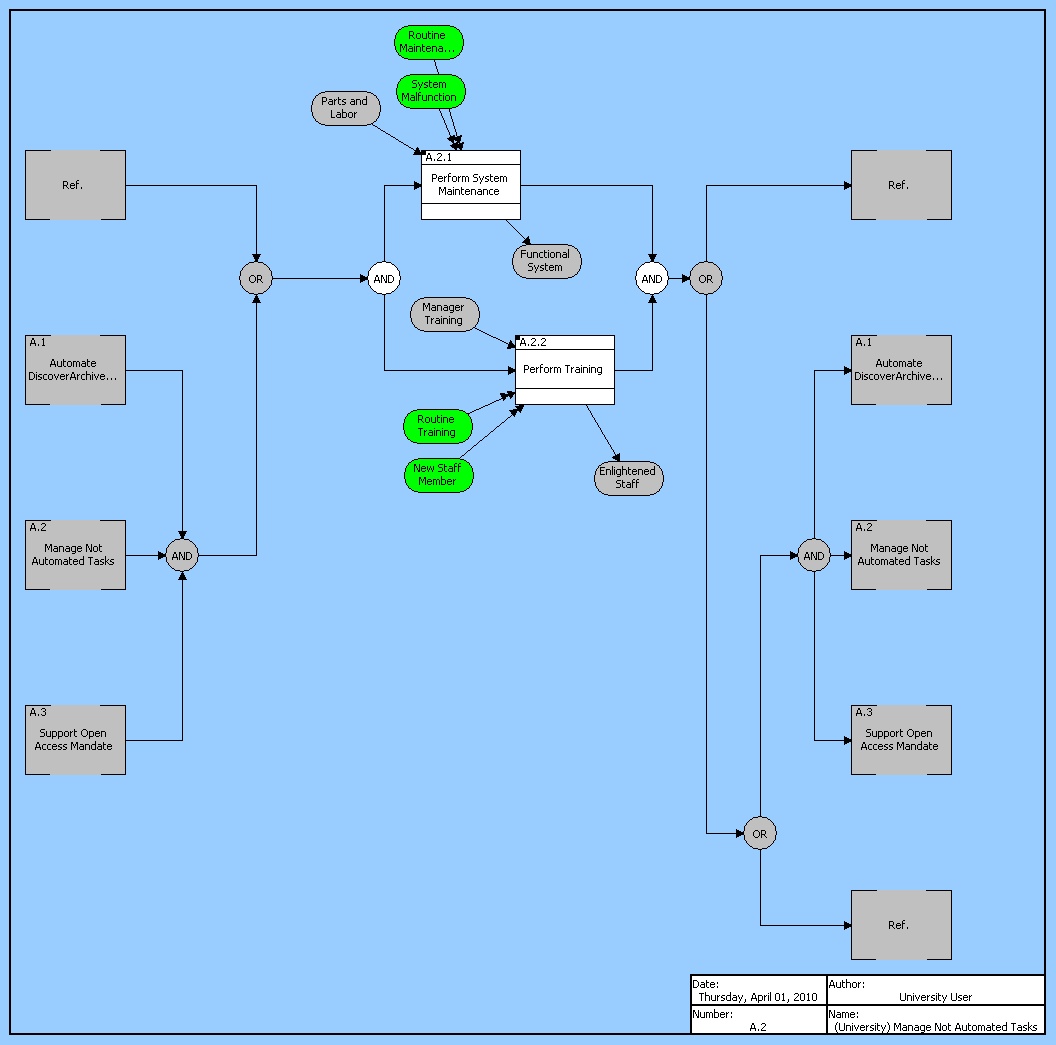


Figure 37

The functions in figure 37 show how the system impacts some of the human components. It is simultaneously maintaining the physical components of the system while providing training for new staff members. Managers input training given the triggers of new staff or recurring training call the function to take that input and produce a more educated staff member.

## **7.5 Proposed System Physical and Operational Architecture**

Physical Architectures

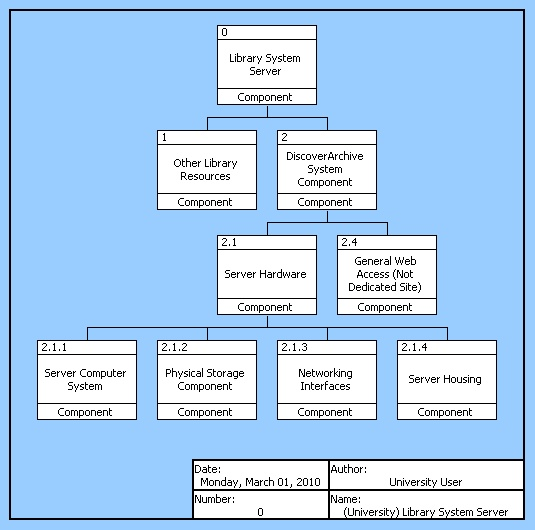


Figure 38

Figure 38 above shows the generic physical architecture for the incremental system. As you can see, the system provides a means for basic automation of fundamental manager tasks but does not automate all system requirements. This physical architecture will not require a large expansion of existing library resources. There would be a dedicated unique storage server set up to host DiscoverArchive files, but it would use the existing website infrastructure and have limited external interfacing.

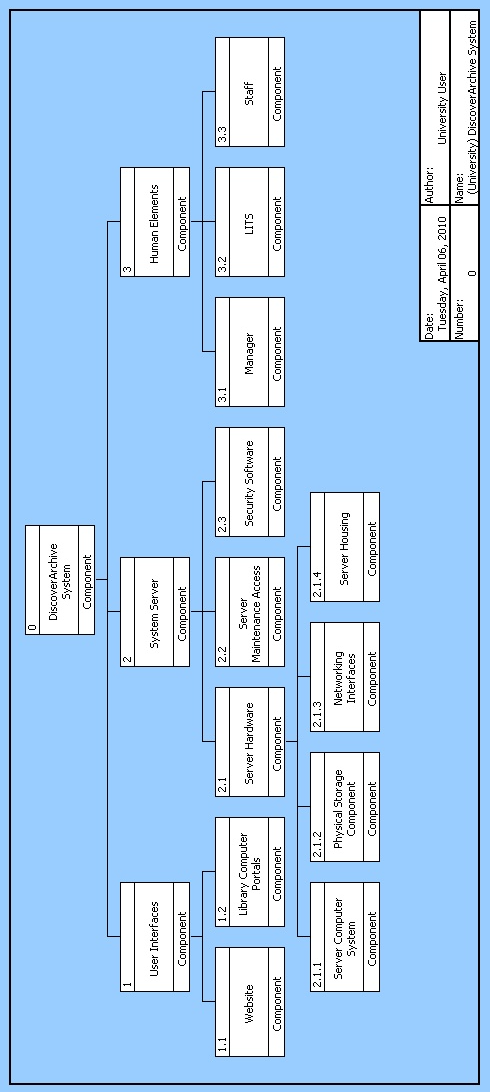


Figure 39

Figure 39 shows an expanded physical architecture that represents the progressive system implementation. Although the physical appearance of the two system architectures does not differ greatly, the actual capabilities of the system do. For example, the progressive system includes a revamped, fully independent and autonomous website that resides on a much more robust dedicated server. These two traits alone provide for a majority of the increased capability of the system to handle an exponentially expanded document flow. Also, smaller attributes such as increased network interfacing capability with outside peer networks and iTunes U to allow greater access to resources is included. Another difference is that the progressive system contains defined human elements. This is because the progressive system will include tasks that simply cannot be automated, or that would not be practical to automate. The functional hierarchy describes these tasks as they relate to the main human components of the manager, LITS, and general staff.

Morphological Representation of Selected System Components

Below is a morphological box that shows the various options available for critical system components. The general trend in the difference between the components for the incremental and progressive systems is that the progressive system tends to demand new, dedicated software and hardware that is capable of handling a large increase in dataflow and users. If the open access initiative is mandated, DiscoverArchive will experience a massive increase in both of these that will without a doubt overrun current system resources. This means that the progressive system must be built to handle all of the implications of a mandate, whether this means simultaneous users, data transfer bandwidth or access permissions

|  |  |  |  |
| --- | --- | --- | --- |
| **System** | **Website** | **Physical Storage** | **Security Software** |
| Incremental | The incremental system will retain the existing website hosted on the current Library server | The incremental system will, by nature, have a lower limit on the data that it can handle, thus demanding a smaller physical storage capability | The existing library server security software will most likely be sufficient for safeguarding the incremental system |
| Progressive | The progressive system will demand a new streamlined and re-engineered website that is independent and capable of handling increased users and document flwo | The progressive system will require a massive dedicated storage system to handle the increased research document flow of all media types | The progressive system will contain a massive amount of research data for public use. This data must be secured to ensure that it cannot be tampered with, and the system will have to track a large number of user permissions demanding robust security software |

Operational Architecture

|  |  |
| --- | --- |
| **Component** | **Function** |
| 1.1 Website | A.1 Automate DiscoverArchive Process  A.1.1.1 Gather Uploaded Research  A.1.1.2 Create Metadata Templates  A.1.1.3.1 Cross Reference with publisher policy  A.1.1.3.2 Create Contract Addendum  A.1.1.4 Assign cataloger to research  A.1.2.2 Provide User Interface  A.3 Support Open Access Mandate  A.3.1 Facilitate Registration |
| 1.2 Library Computer Portals | A.1.2.2 Provide User Interface |
| 2.1.1 Server Computer | A.1.1 Archive Research  A.1.2 Host website  A.1.2.2.1 Accept research uploads  A.1.2.2.2 Provide research downloads  A.1.3 Digitize Research  A.1.4 Archive iTunesU content  A.1.4.3 Publish iTunesU content to archive |
| 2.1.2 Physical Storage | A.1.1.4.2 Publish to DiscoverArchive |
| 2.1.3 Networking Interface | A.1.2.1 Connect to peer networks  A.1.3.1 Coordinate with third party vendors  A.1.3.2 Acquire Copyright permissions  A.1.4.1 Download iTunesU content |
| 2.1.4 Server Housing | A.0 System Exists |
| 2.2 Server Maintenance Access | A.2.1 Perform system maintenance  A.2.1.1 Respond to server malfunctions  A.2.1.2 Perform Routine Maintenance checks |
| 2.3 Security Software | A.2.1.1.1 Notify User of Malfunction  A.3.1 Facilitate registration  A.3.1.2 Process Registration Request |
| 3 Human Elements | A.2 Manage not automated tasks |
| 3.1 Manager | A.2.2 Perform training  A.2.2.2Provide Administrator training |
| 3.2 LITS | A.2.2.1 Provide User training  A.2.2.1.1 Create Submission tutorial  A.2.2.1.2 Create query and retrieval tutorial |
| 3.3 General Staff | A.2.2.3.1 Schedule student training  A.2.2.3.2 Maintain Digitization Workflow |

The operational architecture for the proposed progressive system shows that a majority of tasks fall upon the automation capabilities of the system website and server. It is difficult to assign certain functions, such as the posting of approved documents to the archive, to a single component because many require the interworking of multiple components (in this case the server computer would add access to a given document stored in the physical storage to the website). For this reason some functions occur multiple times in the hierarchy.

# 8 Testing and Evaluation

## 8.1 Proposed System Metrics

*Incremental System Changes*

The incremental system changes will work by automating some of the tasks that are currently handled by the system manager. These tasks will be automated through the use of a simple website. The website will serve as an upload portal for university researchers. This will automate the system manager’s responsibilities for gathering research from around the university. While the manager will still need to promote the use of this website, it should increase the amount of research flowing into the DiscoverArchive system. Thus, the incremental system changes will not create a fully automated DiscoverArchive system, but rather, it should improve the open access success of the system. The best technical performance measure to track the success of these changes would be to follow the amount of research submitted through the upload website. Currently the system manager uploads approximately 100 research documents each month. A good target goal for this technical performance measure would be to double this upload rate to 200 research documents each month. This should be achievable, especially if the system manager continues working on acquiring research in conjunction with the upload website.

*Progressive System Changes*

The progressive system changes involve automating nearly all aspects of the existing system. This would require a more complete development of the research upload website. The idea being that the upload website would also require uploaded research to be in a proper format with metadata that is entered by the researcher. This would then allow the website to direct all uploaded research directly to catalogers for publication to DiscoverArchive while completely bypassing the system manager. Additionally, the progressive system changes would include the designing of a software package that could automatically harvest files from iTunesU and other sources, create metadata from them, and then direct them to catalogers for open access publication. This is currently one of the most time consuming tasks for the system manager so its automation would free up a large amount of time. The progressive system changes also create more structure for supporting the university mandate toward open access and regulating the management tasks of the system. Because other automation in the system will free up a significant amount of time for the manager, this time could then be used for promoting the open access mandate and creating training information for researcher that are uploading information to the system. If the progressive system changes are implemented, the amount of researched published to open access should jump dramatically, particularly with a university mandate. For this reason, the TPM’s for the progressive system changes should set goals for the amount of research being published to DiscoverArchive. A reasonable goal for published documents through the upload website could be 400 uploads per month, while information harvested from iTunesU and other sources should increase by several factors. Additional TPM’s could track the number of university researchers that are registering for DiscoverArchive access permission. With the passing of a university mandate, goals for this could be 50% of researchers within the first 3 months, 75% of researchers within the first 6 months, 85% of researchers within the first 9 months, and 95-100% of researchers within the first year of passing the university mandate.

## 8.2 Qualification Matrix

*Incremental System Changes*

The primary TPM for the incremental system changes is to reach a goal of 200 research publications to DiscoverArchive each month. This should be tracked on two fronts. First, the number of documents that are uploaded through the website upload portal should be recorded to determine if it is being used adequately. If the numbers from this validation are not encouraging, some marketing work may need to be done to encourage researchers to submit files through this method. Second, the amount of published research that is gathered by the system manager should still be tracked. If this number diminishes or is not satisfactory, the website upload portal needs to become the main focus for research gathering, an improvement needs to be made to the system managers gathering methods, or both.

*Progressive System Changes*

Similar to the incremental system changes section, the TPM’s for the progressive system changes should also be monitored. In the previous section, the TPM’s for the progressive system changes were identified, and in this section, the verification of these metrics will be discussed.

|  |  |  |
| --- | --- | --- |
| **TPM** | **Verification Process** | **Resolution if not Verified** |
| 400 uploads per month from website upload portal | Tracking from LITS to determine how many times uploads are made through the portal each month | -Improve training of researchers for uploading  -Market the DiscoverAccess upload portal to researchers |
| Harvest 5x more files from iTunesU than existing system | Compare the amount of files harvested by the new software to past data from existing system | -Check for software bugs  -Improve software  -Run the software more frequently |
| Train all DiscoverArchive undergraduate assistants 1 month after system changes | Hold 2 separate verification seminars to ensure that all assistants have been adequately trained | -Find more undergraduate assistants that will attend training  -Hold training sessions |
| Register university researchers (50% in 3 mos., 75% in 6 mos., 85% in 9 mos., and 95-100% in 1 yr.) | Cross reference the list of registered researchers with a list of researchers that should be registered | -Improve registration marketing efforts  -Send notification emails to those that have not registered  -Create incentives for those that do register |

# 9. Risk Management

## 9.1 Risk Identification and Analysis

The mission of Discover Archive calls for both preservation and dissemination of digital works. Subsystems that accomplish each of these requirements introduce risks into the system. For preservation, the broadest risk is failure of the system in actually preserving digital materials. Any areas where the system could lose or render data useless pose risks of this nature:

-data transfer between components of the system

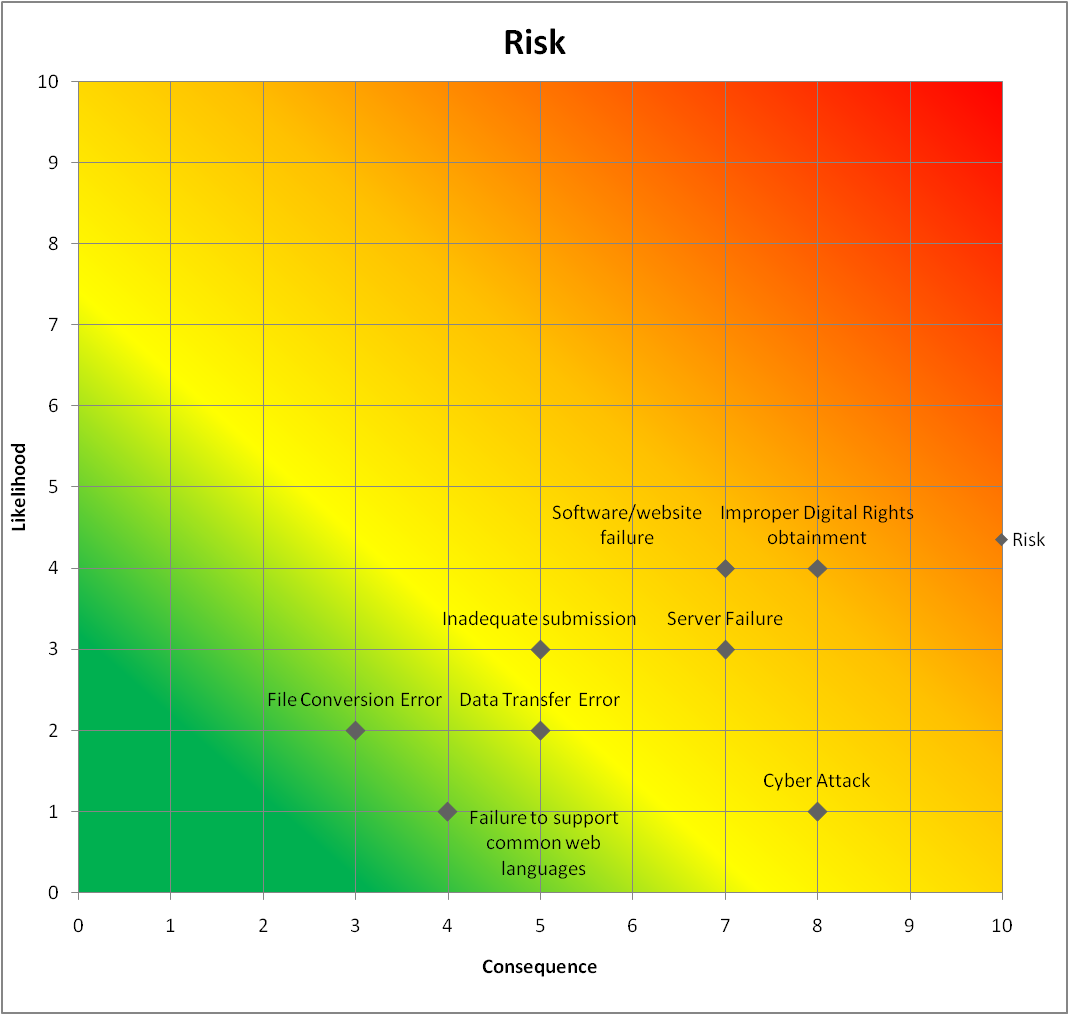
-conversion of files between different formats

-server failure

-software error

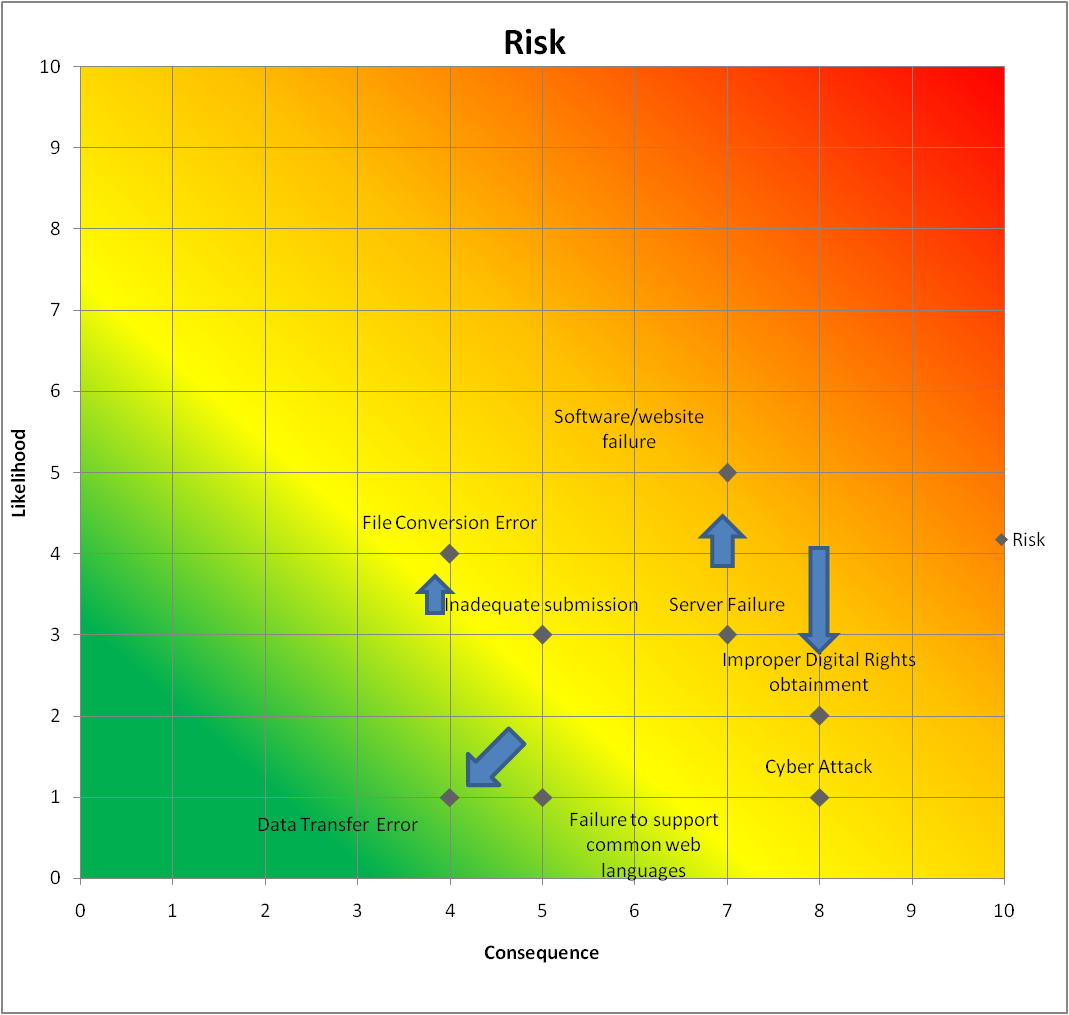
-cyber attack

Dissemination of the data on the other hand, introduces primarily legal risks that could prevent the system from accomplishing its mission, although there are some technical risks as well. Legal risks stem from the possibility of lawsuits over digital rights which could take up large amounts of the human capital in the DiscoverArchive system which would in turn pose even more risk to the system (especially the incremental system where human capital is employed in a more direct manner). Lawsuits would stem from incorrect obtainment, interpretation, or implementation of digital rights. Disseminating information without the proper rights management would trigger such lawsuits. The technical risks to the dissemination mission include the risk of incompatibility with other software units such as Google crawlers or basic web languages such as html. Also, contributors to the database must be closely monitored so that they do not compromise the credibility of the archive by submitting research of an inadequate caliber.



9.2 Risk Matrix

Above, the diagram shows that risk areas of concern are improper digital rights management and software/website failure. Since the human component has to handle the digital rights process there is a higher likelihood of error, and the concequences are great as a lack of digital rights undermines the mission of the system. Software/website failure also undermines the dissemination mission and with a new any new system there is likelihood for error



This diagram shows the movement of the risks when they are assessed in the context of the progressive system. With a more complex website, likelihood of a bug increases. Also, file conversion error likelihood increases as the process is now more automated and the automated system might have a harder time recognizing an incorrect conversion. However, the biggest risk, improper digital rights obtainment drops in likelihood as does data transfer error.

## 9.3 Risk Mitigation Strategy

*Incremental System*

The mitigation strategy will be to implement a TPM on the digital rights. After the system has checked the publisher database, and sent the pre-prepared information over to the cataloger, the cataloger will have one day to verify to ensure that no further action is taken on the document without having first passed a pair of human eyes. Also, the website and servers should be backed up offline in secure location

*Progressive System*

The mitigation strategy on the progressive system focuses on the software and website failure. In order to reduce the likelihood, the system will undergo significant lab and beta testing. Furthermore, the system is designed to support user training, which should help reduce the consequences of a user prompting the software to execute something harmful to the system or to a particular document.

# 10. Technical Program Planning, Implementation, and Control

## 10.1 Work Breakdown Structure



The work break down structure outlines the tasks needed to take the system from its current state to the intended new system. This visual shows the flow in a top to bottom manner, with three areas of tasks across the top. Website, Human Staff, and Policy/User Communication task areas each have specific tasks that need to accomplish to get to the first level functions listed on the bottom row. There, the new system is fully implemented and the functions will perform in order to accomplish the system requirements and objectives.

## 10.2 Project Schedule

The Critical Path outlined below shows the path of tasks that are dependent on each other-that is if any of these tasks are delayed, the final completion date will be delayed. Many of the tasks are sequential, thus the critical path is very similar to the overall task path.

## 10.3 Budget Overview

The following is a description of the origins for our budget predictions. These were rather vague, but we believe that we will have access to more specific information about the Duke and Kansas Universities implementations by the end of the week.

* **Develop New Interface**: this section of the design implementation will flow from the SEMP outline we have created this semester. The planners will take a couple of days to go over the physical requirements we outline and choose actual components, and then the software team will then write the hardware and develop the website. After this is complete the hardware will be purchased. This implementation level’s budgeting is based on rough programming hourly salaries as well as library tech wages and actual physical hardware costs.
* **Interface Implementation & System Verification**: These sections are entirely composed of hourly wages and completion time estimates for the professional and library techs that will physically install the system. For example, the verification process is currently budgeted for one person at an assumed hourly rate of $30/hr to run through all system tasks. This could easily be expanded to two or more testers.
* **Staff Training**: The staff training levels are self-explanatory, and the budget accounts for special techs to train the library staff but at this point assumes that library staff are under normal payroll and would not be covered by the project implementation budget. The training ends with a day of Strategic Planning to cover how to use DiscoverArchive as it applies to the broad goals of open access.
* **Incremental Activation**: The implementation process ends with a 12 day incremental activation period in which the system is slowly opened for use. First registration will take place, and then file uploading with be activated followed by full university access and finally full public open access. The reason for the incremental activation is that the system will be handling a massive data flow, and we will want to be able to perform incremental trouble shooting without the full intended data flow. The budgeting for this section accounts for a tech to be available for maintenance and troubleshooting.
* **Total Cost For Incremental**: $8,300
* **Total Cost for Progressive**: $22,850

# 11 Recommendation and Key Issues/Future Directions

## 11.1 Analysis of Alternative Approaches

*Incremental Changes*

**Pros**

* Increased research intake
* Moderate automation of tasks
* Easily implemented

**Cons**

* Could not handle a full open access mandate
* Still requires manager to handle large amounts of data
* Does not address alternative media type handling

*Progressive Changes*

**Pros**

* Automates all research intake
* Automates iTunesU data handling
* Creates contract addendums for researchers and publishers
* Relieves manager from metadata tasks

**Cons**

* Requires large website redesign
* Difficult/expensive to implement
* May require research faculty training

## 11.2 Outstanding Issues

With both the incrementally and progressively changed proposed systems, there are several issues that need to be addressed. Answering these questions will aid in making a decision about how to proceed in upgrading the system.

* Will the university make an open access mandate? If so, when?
* In the event of an open access mandate, how much web traffic will the DiscoverArchive receive, and will the system be able to handle this traffic?
* Is it feasible and within the DiscoverArchive budget to proceed with a large website and software design process?
* Will research faculty members be compliant with an upgraded DiscoverArchive system?

## 11.3 Future Directions

Future directions for this system revolve mainly around whatever direction the open access movement takes. If it continues to gain popularity among research institutions and academic universities than more regulations will probably be defined. The relationship with publishers is certainly going to be a dynamic one over the next few years as they fight to maintain some revenue stream. These changes will not have much of an effect on the physical system implementation proposed by our analysis, but an effective follow project would be to reevaluate the goals and capabilities of DiscoverArchive as they apply to open access. This project scope is extremely vague, but the truth is the movement could go anywhere. Our personal feeling is that it will continue to strengthen to the point where some central database, possibly an addition to the Google Scholar search engine, will be created. At that point DiscoverArchive will need to be re-evaluated to ensure that it functions within the scope of the larger movement.

# 12 Discussion of Selected Approach

After being presented with both the incremental and progressive DiscoverArchive system overhauls, the Vanderbilt Digital Archive managers engaged in an active discussion that ultimately resulted in the choice to pursue the incremental system. Their decision was not unexpected, because we have a good understanding of what kinds of initiatives their current resources will permit, and the progressive system was designed under the assumption of increased funding succeeding an Open Access mandate from the university. Ultimately the incremental system is a better option for upgrading Vanderbilt’s Digital Archive at the moment because it will allow the university library system to continue to follow emerging the open access movement without stepping outside of it resources.

The reason for performing this systems engineering analysis is to discover why the DiscoverArchive is not fulfilling its purpose and to map out how it can be adjusted to do so. To do this we began by defining the purpose of the archive as being an open source of information for not only university members but also the general public (through Google Scholar and other interfaces). When considering this definition, it is quite apparent that the archive in its current instance is far from capable of carrying this weight, mainly due to the fact that it relies on a limited number of human components to carry out day to day operations. This means that the document flow is severely restricted, and we immediately focused on this as a basis upon which we built our proposed systems. Therefore, both system are fundamentally similar and really only differ in their physical implementations. For example, the progressive system incorporates an entirely new state of the art server system that would handle the document flow anticipated with a full-scale open access archive whereas the incremental system simply draws more resources from the existing library system.

From this systems engineering management plan one can see that it makes more sense to pursue the incremental system when there is no University mandate for open access. This choice however does not rule out the possibility of an eventual installation of the progressive system, and in fact lays a foundation to facilitate it. The incremental system will include a new website and more robust automated server software that can easily be ported to a progressive system. We fully agree with the Digital Archive managers’ decision to go with the incremental system for now because it will sufficiently boost system performance to be capable of handling increased document flow. It will leave the door open for future expansion if a university wide mandate for open access is passed, in which case the mandate would bring the necessary funding to implement the progressive system on top of the framework laid by the incremental system. For that reason, detailed knowledge of both the incremental system and progressive system will add value to the entire DiscoverArchive project.

# 13 References

arxive.org -- physics, math, comp science, quant analysis open access

www.oaklist.qut.edu.au -- publisher database

http://www.sherpa.ac.uk/romeo -- SherpaRoMEO publisher database (red, yellow, green)

http://www.opendoar.org  -- open access repository database (has over 1500 open access repositories)

* Carl Lagoze and Herbert Van de Sompel (2001). "[The Open Archives Initiative: Building a Low-Barrier Interoperability Framework](http://www.openarchives.org/documents/jcdl2001-oai.pdf)". *Proceedings of the first ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'01)*. pp. 54–62.
* Lynch, Clifford A. (2001). ["Metadata harvesting and the open archives initiative"](http://www.arl.org/resources/pubs/br/br217/br217mhp.shtml). ARL Bimonthly Report 217.
* Frank McCown, Xiaoming Liu, Michael L. Nelson and Mohammed Zubair (March/April 2006). "[Search Engine Coverage of the OAI-PMH Corpus](http://library.lanl.gov/cgi-bin/getfile?LA-UR-05-9158.pdf)". *IEEE Internet Computing* **10** (2): 66–73.
* Herbert Van de Sompel and Carl Lagoze (2000). "[The Santa Fe Convention of the Open Archives Initiative](http://dx.doi.org/10.1045/february2000-vandesompel-oai)". *D-Lib Magazine* **6** (2). [doi](http://en.wikipedia.org/wiki/Digital_object_identifier):[10.1045/february2000-vandesompel-oai](http://dx.doi.org/10.1045%2Ffebruary2000-vandesompel-oai).
* Herbert Van de Sompel, Jeffrey A. Young, and Thomas B. Hickey (2003). "[Using the OAI-PMH ... Differently](http://dx.doi.org/10.1045/july2003-young)". *D-Lib Magazine* **9** (7/8). [doi](http://en.wikipedia.org/wiki/Digital_object_identifier):[10.1045/july2003-young](http://dx.doi.org/10.1045%2Fjuly2003-young).

# 14 Appendices and Applicable Documents

## 1: Project Description

**Organization:** Vanderbilt Jean and Alexander Heard Library – DiscoverArchive ([discoverarchive.vanderbilt.edu](http://discoverarchive.vanderbilt.edu/))

**Project:** Implementation of Open Access service on Vanderbilt campus

***System vision***

The Jean and Alexander Heard Library system provides digital preservation and access services to the Vanderbilt campus through a program called DiscoverArchive ([discoverarchive.vanderbilt.edu](http://discoverarchive.vanderbilt.edu/)). Digital content produced by Vanderbilt faculty, staff and students is collected, described and opened to the public to be searched through the interface as well as by Google and other search services.

Academic institutions worldwide are interested in the concept of providing Open Access to the research output of faculty. MIT, Harvard, Stanford School of Education, University of Kansas and Trinity University are all examples of institutions that have established an Open Access policy of depositing research into an institutional repository like Vanderbilt’s DiscoverArchive. While Vanderbilt faculty have not yet adopted an Open Access policy, some faculty have decided to contribute their research to the repository.

For more info on Open Access see <http://www.earlham.edu/~peters/fos/overview.htm>

***System objective***

Develop a system and workflow to implement Open Access service at Vanderbilt University making this process as easy as possible.

Areas to look at would be how to easily generate publisher agreements for faculty that would allow for copies of research to be collected at the university and the best way to collect and deposit research into the DiscoverArchive system considering current workflows.

***How the system would be used***

This system would be used by faculty as a way to independently generate their own addendums to request open access from publishers and a way for them to easily get the digital copy to the library for processing. Regardless of whether a policy of Open Access is adopted by faculty, this process will be put into use for those who choose to participate.

Contact:

Ronee L. Francis

DiscoverArchive Manager

[r.francis@vanderbilt.edu](mailto:r.francis@vanderbilt.edu)

(615)322-2807

## 2: Interview Transcript

Interview Log

Date: 4/7/2010

Interview with:Ronee Francis, Vanderbilt University, Digital Collections Archivist

Q: What changes would you like us to make to our current model of the existing system?

*A: Change ITS to LITS; focus on the library as a stakeholder; also the end user as a stakeholder*

Q: Is there anything that you would like us to research regarding the existing system?

*A: Determine the number of faculty; assess the amount of research being published from Vanderbilt*

Q: Could you please explain further how the contract addendum process works?

*A: Addendum process must be completed before open access publishing can occur; when a document is submitted, the researcher must then take the addendum, fill it out, send it to the publisher, receive it from publisher before the doc can be published to open access*

Q: Do you have any methods in place for tracking web traffic on the DA website?

*A: Yes we use Google Analytics to track website traffic. The average monthly traffic is approximately 5000 vists and 21000 page views.*

## 3: Interview Transcript

Interview Log

Date: 3/25/2010

Interview with:Ronee Francis, Vanderbilt University, Digital Collections Archivist

Q: What are your basic repository taskz?

*A: Basic tasks include creating collectionz, metadata templates, and assigning files to catalogers for archiving.*

Q: What is your interaction with graduate students?

*A: Includes coordination with faculty and graduate research programs.*

Q: Are there any other tasks along these lines that you would like us to model?

*A: Archiving research that isn’t in digital format can be challenging because of the conversion process.*

Q: What are your responsibilities to the system outside of archiving?

*A: Responsibilities toward managing work schedulez and training programs.*

Q: You had mentioned iTunesU, do you pull information from this system?

*A: Yes and it is very time consuming to create this metadata.*

## 4: Interview Transcript

Email Transcript

Date: 3/16/2010

Benchmark organization: Duke University with Kevin Smith, Scholarly Communications Officer

Hello Mr. Smith,   My name is Nick and I'm a senior Engineering major at Vanderbilt and I'm in a Systems Engineering class this semester. Part of our class involves applying course concepts to a real world scenario and for our group project, we are analyzing Vanderbilt's Open Access Policy as it applies to the university's Discover Archive digital repository. One of the staff at our university library passed along your contact information and we were wondering if you could help us with a few questions, since we understand you are looking at Duke's Open Access policy.   1) How does the faculty get the addendum?  -Does the faculty member add the research to the repository themselves or does the repository manager (RM)? If it is the RM, how do they get the addendum and the PDF to the RM? All in a way this is super easy for faculty.

2) Also, how is Duke's repository structured? Is faculty research put into one general collection within each department or a personalized collection. Does each faculty member have a dedicated collection of their research, or is it aggregated into a subject or department wide collection.

Lastly, if you know of someone who might be able to add even more insight, we'd love to have their contact information as well.

Thanks for your time!  Bestregards, Nick Williams

**RESPONSE:**

Hi Nick,

I am happy to help, but you should know that we are at a stage were the open access policy has been proposed to our faculty but not yet voted on.  The repository we currently have in place is mainly used for electronic theses and dissertations at this point.  So much of what I have to say is theoretical, or, perhaps it is more accurate to say, aspirational.

At this time we do not plan to use a specific addendum.  As we tell faculty, most journals already allow self-archiving in an institutional repository.  For another group, where the journal does not permit self-archiving and the author does not want to negotiate, we expect that the policy/license will be waived.  Thus there should only be a small group for whom an addendum would be valuable.  Even in those cases, I doubt the utility of submitting an addenda, since I think publishers tend to reject them out of hand.  Instead, I would advocate for a direct request for the needed rights, so that the publisher could write them into their regular contract.  For some reason, publishers seem more comfortable and cooperative when that course is pursued, even when the end result is the same as accepting an addenda.

We are discussing ways in which repository deposit will be managed, and I imagine that several different models will exist.  Certainly faculty will be able to deposit directly themselves, but I doubt that model will actually be used very often.  We expect that a repository manager or, more likely, a small group of librarians that include subject specialists will get the citation and, where possible, the PDF.  In so far as possible we would like to make this whole process reactive for the faculty, so that they need only say OK to a proposal that a set of their articles be deposited.

For the small amount of faculty scholarship that is currently in our repository, communities have been created for academic departments, but that could certainly change.  We are in conversations, for example, about the design of a new system for faculty to report their yearly publications as part of the promotion and tenure process; if we could link repository deposit to that reporting process, it is very possible that the basic unit within the repository would be individual faculty members.  This would also facilitate generating individual web profiles with links to full-text for each faculty member.

For more information, as well as to tap into his vast knowledge of different repository systems, I suggest that you contact Paolo Mangiafico, who is Duke’s “Director of Digital Information Strategy” appointed by our Provost.  His email is [paolo.mangiafico@duke.edu](mailto:paolo.mangiafico@duke.edu), and I have copied him here by way of introduction.

Good luck with your project.

Sincerely,

               Kevin Smith

Kevin L. Smith, M.L.S., J.D.

Scholarly Communications Officer

Duke University, Perkins Library

P.O. Box 90193

Durham, NC 27708

919-668-4451

[kevin.l.smith@duke.edu](mailto:kevin.l.smith@duke.edu)