

2013

University of North Carolina Wilmington
Master of Science in
Computer Science and Information Systems
Proceedings

<https://csbapp.uncw.edu/mscsis>

ArcGIS Server: An Application Migration, Application Template Construction
and Server Analysis & Design for the
University of North Carolina Wilmington Department of Geography & Geology

Aaron Caldwell

A Capstone Project Submitted to the
University of North Carolina Wilmington in Partial Fulfillment of
the Requirements for the Degree of
Master of Science

Department of Computer Science
Department of Information Systems and Operations Management
University of North Carolina Wilmington

2013

Advisory Committee

Dr. Douglas Kline

Dr. Joanne Halls

Dr. Ron Vetter

Dr. Brian Reinicke

ABSTRACT

What critical components of a GIS system must be monitored in the course of migrating from an older GIS Server architecture to another that's nearly 10 years newer? In this paper, we examine the requirements at the system and application levels as we assist UNCW's Department of Geography & Geology in a migration from ArcGIS Server versions 9.2 and 10.0 to version 10.1. We examine the components that change, as well as those that stay the same and determine their overall impact on software development. Finally we look at application design and determine how much can be salvaged from critical older applications as we migrate them to the new architecture, and determine how application development will change from this point forward.

Contents

1.0 Introduction	1
1.1 Geographic Information Systems	1
1.1.1 Definition of GIS	1
1.1.2 Origins of GIS	1
1.2 Esri	3
1.2.1 Esri Company Overview	3
1.2.2 ArcGIS Software	3
1.3 GIS in the World	4
1.4 GIS in Education.....	5
1.5 Project Justification	5
1.6 Project Intent	6
1.7 Summary	7
2. Current Technology and Literature Review	7
2.1 Server-based Computing.....	7
2.1.1 Thin Clients	8
2.1.2 Thick (Fat) Clients.....	9
2.2 GIS Servers	9
2.2.1 Open-Source and/or Free GIS Server Solutions	10
2.2.2 Enterprise GIS Solutions	11
2.3 Analysis & Design.....	11
2.3.1 Object-Oriented Analysis and Design	12
2.3.2 Structured Analysis and Design Technique	12
2.3.3 Service-Oriented Analysis and Design.....	13
2.4 ArcGIS Performance and Scalability.....	13
2.3.1 GIS Server Terminology	13
2.3.2 GIS Server Performance and Scalability	16

2.5 Web Portal Design Specific to UNCW	19
3. Project Methodology	21
3.1 Design of Research.....	21
3.1.1 Tools & Applications.....	21
3.1.2 Timeline	21
3.1.3 Outcome Measures	22
3.2 Project Execution	24
3.2.1 System Analysis.....	24
3.2.2 Service & Application Migration.....	50
3.2.4 Sample Application.....	64
3.2.5 Web Portal Update	68
4. Results	69
4.1 ArcGIS Server 10.1 Install.....	69
4.2 Application Migration	70
4.2.1 Oculina.....	70
4.2.2 Riverrun.....	72
4.3 Sample GIS Application.....	74
4.4 Portal Update.....	75
4.5 System Design.....	78
5. Conclusions, Limitations & Recommendations	82
5.1 Application Development.....	82
5.2 Analysis & Design.....	84
5.3 Limitations	84
6. Future Work	84
7. References.....	85
Appendix A: Project Charter As Presented To the committee On November 9, 2012.....	89
Table of Contents	90

1. Project Purpose	92
2. PROJECT EXECUTIVE SUMMARY	92
3. PROJECT RATIONALE	93
4. PROJECT SCOPE.....	94
4.1 Goals and Objectives	94
4.2 Departmental Statements of Work (SOW)	94
4.3 Key Stakeholders.....	95
4.4 Project Deliverables.....	96
4.5 Project Estimated Duration	96
4.6 Deliverables Out of Scope.....	97
5. PROJECT CONDITIONS	98
5.1 Project Assumptions	98
5.2 Project Risks	98
5.3 Project Constraints	99
6.0 PROJECT STRUCTURE APPROACH.....	100
7.0 PROJECT TEAM ORGANIZATION PLANS	100
Appendix B: Project Charter As Revised to Reflect Project Changes and Updates Agreed to By Project Owner and CLient As Of October, 2013.....	102
Table of Contents	103
1. Project Purpose	105
2. PROJECT EXECUTIVE SUMMARY	105
3. PROJECT RATIONALE	106
4. PROJECT SCOPE.....	107
4.1 Goals and Objectives	107
4.2 Departmental Statements of Work (SOW)	108
4.3 Key Stakeholders.....	108
4.4 Project Deliverables.....	109

4.5 Project Estimated Duration	110
4.6 Deliverables Out of Scope.....	110
5. PROJECT CONDITIONS.....	111
5.1 Project Assumptions	111
5.2 Project Risks	111
5.3 Project Constraints	112
6.0 PROJECT STRUCTURE APPROACH.....	112
7.0 PROJECT TEAM ORGANIZATION PLANS.....	113
Appendix C: Requirements that came out of project proposal	114
Appendix D: Map Service Elements Identified for Potential Migration	117
Appendix E: Web Application Functionality and Elements Checklist.....	120
Appendix F: Wireframes for Migrated Applications.....	125
Appendix G: Original Business Process Model & Notation (BPMN) Diagram As Proposed for Sample Service (Later Replaced by Application Workflow in Appendix N)	127
Appendix H: Actor Diagrams	128
Appendix I: ArcGIS 9.2 Server Setup	131
Appendix J: ArcGIS 10.0 Server Setup.....	133
Appendix K: ArcGIS 10.1 Server Setup	135
Appendix L: ArcGIS 9.2 Application Creation Work Flow	137
Appendix M: ArcGIS 10.0 Application Creation Work Flow	138
Appendix N: ArcGIS 10.1 Application Creation Work Flow	139
Appendix O: Results of ArcGIS 10.1 Server Host Read/Write Tests to Local vs. NAS Drive	140
Appendix P: Map Service Creation Instructions	141
Appendix Q: Web Map Creation Instructions.....	149
Appendix R: Instructions for ArcGIS.com Application Creation.....	155
Appendix S: Instructions for JavaScript Application Creation from a Template.....	159
Appendix T: Tools Used on this Project	166
Appendix U: Logging into ArcGIS Server 10.1 As Admin.....	168
Appendix V: Best Practices	169
Appendix W: Deploying JavaScript Applications	171

1.0 INTRODUCTION

1.1 Geographic Information Systems

1.1.1 Definition of GIS

There are three fundamental requirements for determining a geospatial position through use of a GIS: a datum- a reference coordinate system from which geographic measurements can be made, a geographic coordinate system, and a two-dimension representation of the curved surface of the earth (map, computer screen, etc.).

Cartographers use this information to represent geographic features and their relationships to each other. A Geographic Information System (GIS) shall be defined as a computer-based information system used to store, process and modify geospatial data. A geospatial database serves as the backbone for a GIS storing large amounts of data. Interaction with stored data can occur using a variety of different interfaces including command line, web browsers and specialized software packages (Worboys, Duckham 2004). The origins of GIS are diverse and can be traced back to several sources starting in the 1960s.

1.1.2 Origins of GIS

The Harvard Graduate School of Design first produced a mapping package called SYMAP in 1964. SYMAP used line printers to represent different geographic features in raster format using combinations of characters and symbols to represent geographic features. Users could visualize map data by printing it from a GIS using a line printer for output. Though the visual quality of the maps was poor, SYMAP represented the earliest proof-of-concept that computers could be used to produce maps. At about the same time, two of the earliest Geographic Information Systems were implemented at Harvard Labs: GRID and ODYSSEY in 1964 and 1967 respectively. GRID represented one of the earliest attempts at a full-fledged cartography and mapping application. Its technology paved the way for more modern GIS applications in the years to come. ODYSSEY was the first modern vector GIS and would set the stage for advances in vector-based GIS technology (Brimicombe 2010).

Sparked by initial interest in SYMAP, several other GIS packages were developed over the next 10 years. Some packages sought to extend SYMAP's functionality while others built upon similar concepts implemented in standalone products. Several of these packages were referred to as the "Harvard Packages". CALFORM extended use of SYMAP by adding a pen plotter, a considerable improvement over using symbols and characters previously used to represent geographical features. SYMVU, another GIS product took map output a step further by producing 3D perspectives of SYMAP data (Brimicombe 2010).

Though several of advances around this time may be attributed to the Harvard Packages, similar advances were occurring in parallel in Canada as well. Between the 1960s and 1970s, the Canadian government sought a better way to manage its country's vast natural resources. In 1966 the Canada Geographic Information System was implemented to map both land-use and and natural resource availability. The development of this GIS brought with it technologies that would be critical to GIS development including optical scanning of maps, raster-to-vector conversion, a spatial database management system, and a method of dividing maps into "tiles" (Brimicombe 2010).

Similarly, at this time the U.S. government was also investigating methods for GIS-based tracking of human population data. In the late 1960s the U.S. Bureau of Census developed the *Dual Independent Map Encoding* scheme as a data management system that allowed insertion of new spatial-relationship data. Whereas earlier datasets consisted of unrelated collections of geographic features represented as lines, the DIME scheme allowed for interrelation between different datasets for interpretation and analysis. Giving geospatial data a defined structure also allowed for comparison of similar datasets for completeness and accuracy (Brimicombe 2010).

Following its origins in the 1960s, the evolution of GIS developed roughly in parallel with the development in the field of Information Systems as a whole. Advances in GIS technology were often limited by the hardware and software available to utilize them. Moore's law states that hardware's price to performance ratio is expected to double every 18 months (Merriam-Webster 2013). Since a GIS is by its nature a very data-intensive system, advances along already established paradigms were limited by the bottleneck of hardware performance.

1.2 Esri

1.2.1 Esri Company Overview

Jack Dangermond studied at the Harvard Laboratory for Computer Graphics and Spatial Analysis before creating the Environmental Systems Research Institute (later just “Esri”) with his wife Laura in 1969. Esri experienced early success when they won government contracts with San Diego County, Maryland and the City of Los Angeles. During this time, Esri also reached out internationally to other businesses with similar aspirations to develop a broad distribution network (Esri 2011).

In 1982, Esri released its first commercial GIS product ARC/INFO. The application combined geographic features presented on a computer display with a database management system. This front and back-end combination allowed a user to view and modify geographic features. The ARC/INFO product line evolved along with a technology shift in the early 1980s towards UNIX workstations and PCs. With this shift, users were able to install ARC/INFO software on a variety of platforms and take advantage of distributed computing and data management. Support for multiple platforms also encouraged developers to produce applications that worked on top of ARC/INFO (Esri 2011).

The 1990s brought explosive growth to Esri and the GIS industry as a whole. Computers were becoming faster and cheaper and networking technology was becoming more convenient and mainstream. The GIS industry also experienced a boon with the advent of geospatial data-capture techniques such as global position systems (GPS) and remote sensing. During this time, Esri introduced ArcView, a desktop GIS software package offering basic GIS functionality that would open up GIS to a wider audience of users. In the late 1990s, Esri re-engineered the original ARC/INFO package to create a scalable platform known as “ArcGIS”; this platform was suitable for use by casual desktop users and scalable up to the enterprise level (Esri 2011).

1.2.2 ArcGIS Software

Today there are a variety of different GIS packages available. A quick internet search will reveal a broad array of paid-for and open-source GIS products available across a variety of platforms. The appropriateness of a given GIS platform for an individual or institution is dependent upon a variety of factors, such as the type of datasets, required functionality, budget and industry trends. Esri remains an industry leader in GIS with ArcGIS Desktop as its flagship product. ArcGIS software is available in a variety of packages with different levels of features and an array of different extensions available. While numerous products are available to analyze data (particularly with databases), ArcGIS Desktop software in particular facilitates the discovery of patterns, relationships and trends in geospatial data that might not be so easily observable otherwise. It also enables quick and easy modeling and analysis of geographic data, the results of which can be displayed via customized maps (Esri 2007).

Whereas ArcGIS Desktop provides capabilities for the authoring of customized maps, ArcGIS Server makes this data distributable via a network or the internet. Users are no longer required to have ArcGIS Desktop installed on their home computers in order to enjoy the benefits of custom mapping applications. Map authors can now create services from their customized maps which can then be consumed by further-customized applications and made accessible to a larger audience. Esri's ArcGIS Server makes it easier to support not only desktop applications but also mobile and Web geospatial applications. Its centralized structure also allows for greater content management from a single source (the server). ArcGIS Server can be installed on a single machine, multiple machines or in the Cloud (Esri 2011).

1.3 GIS in the World

GIS is increasingly being adopted by new industries. Utilities, real estate, government, military and insurance are a few major industries adopting GIS technology. This technology might range anywhere from basic GIS file-reader software to large, interactive, geographically-dispersed data networks. Currently the public sector represents the fastest growing portion of the GIS market accounting for more than one-third of market's total revenue. Regulated industries such as utilities, telecommunications,

transportation and education represent the majority of the public GIS market. (Department of Labor 2004).

1.4 GIS in Education

The Department of Labor (2004) declared the Geospatial Industry as being “high growth” with annual rates of growth at nearly 35% (Geospatial Information & Technology Association). For this reason, the Employment and Training Administration planned to invest over \$8,000,000 in the industry through 2006 (Department of Labor, 2004). With growth in the GIS industry, came a resulting increase in demand for trained GIS professionals. Education in the industry continues to be redefined as employers and educators work together to nail down GIS as a discipline while evolving it to keep up with the simultaneously fast-paced IT industry.

1.5 Project Justification

UNCW’s Department of Geography and Geology strives to keep pace with changing demands on GIS instruction. This instruction is based on the relatively established Geography discipline, counter-balanced by the rapidly evolving field of Information Systems and the broader IT industry. As GIS software and best practices change, GIS instruction must evolve to accommodate these changes.

In 2010, UNCW upgraded to the current version of ArcGIS Advanced 10.1 desktop software. ArcGIS Desktop 10.1 is currently used at approximately 40-50 workstations for undergraduate and graduate student instruction. In conjunction with the desktop software, the department uses ArcGIS Server versions 9.2 and 10.0. Both of these servers remain in operation in order to preserve functionality of older applications which are currently incompatible with the new 10.1 server platform.

Both of the currently configured servers (9.2 and 10.0) are incapable of hosting services created using the newer ArcGIS 10.1 desktop software. One result is that authors of customized maps on the new desktop software are unable to publish services. An additional inconvenience is that users of the ArcGIS 10.1 Desktop software are unable to edit older services currently hosted on the old seThis disconnect between the desktop

software and server software compromises GIS instruction and serves as an inadequate use of UNCW's ArcGIS software site license. In order to keep pace with industry standards, fully utilize the current site license and instruct students using current best practices, UNCW would need to upgrade to ArcGIS 10.1 Enterprise Server and migrate any critical applications from the old servers to 10.1.

1.6 Project Intent

The first portion of this project focused on porting old applications designated as critical for migration to the new server. In a best case scenario, this would require reconciliation of JavaScript and HTML code differences required to republish the applications. A worst-case scenario would require an entire application rebuild for each one. Since each project was different and the applications were built using an older ArcGIS technology (ArcIMS), the level of difficulty for this phase was at first unclear. Therefore the first goal of this project was to determine the required effort for porting/recreating critical ArcGIS 9.2 applications and then to move forward with the best plan for migration. The application migration would benefit UNCW in the following ways:

- Free up UNCW IT system and personnel resources (no longer a need for the 9.2 and 10.0 servers)
- Provide a more consistent GIS server platform for course instruction and faculty/student use
- Better utilize the current site license for ArcGIS Server 10.1

Updated applications would be available for student, faculty and public viewing via the GIS portal following completion of successful migration.

Building on lessons learned from the application migration, a sample application would be posted for student development. There were several available JavaScript templates freely available for download from Esri at the time of this writing, Dr. Halls and I would work together to implement a lightly customized version of one of these templates for further student development.

The third portion of this project, the GIS Portal update, would support access to the ported migrated applications as well as the application template. The Department of Geography and Geology used a web-based portal to access student and instructor-designed map applications. At this point the portal contained links to applications on multiple servers (9.2 and 10.0). Updating the portal with links and descriptions for the new applications would provide a more consistent front-end delivery for both the old and new map applications. The final portion of this project would examine the ArcGIS Server configuration. The ArcGIS Server had been installed using a default setup by UNCW's ITSD. This setup may or may not be sufficient for current/future demands on the server. There are a variety of ways the server might be setup in order to accommodate different hardware, user-traffic, and number of simultaneous clients. With this in mind, this project would include a system analysis & design. The analysis & design would serve not only to verify the current system setup but also to consider alternative configurations for better use of current resources. Exploration of alternative configurations would also serve to anticipate system changes that may be required to meet future needs.

1.7 Summary

The UNCW Department of Geography and Geology needed to complete migration to the new ArcGIS Server in order to maintain adequate system functionality and keep pace with current technology trends. Completion of migration would require:

- Application migration
- Modification and creation of a sample application
- Web Portal update
- Server Configuration Analysis & Design

In order to perform these tasks, it would be necessary to research and examine the associated technologies and assess the different options for project implementation.

2. CURRENT TECHNOLOGY AND LITERATURE REVIEW

2.1 Server-based Computing

Server-based computing (SBC) can be defined as the deployment, management and support of applications executed on a server vs. a client machine. Users on the client side interact with server-based services and applications which then transmit any required data to the client's machine. The client may then interact with the service or application using a defined interface such as Simple Object Access Protocol (SOAP) or Representational State Transfer (REST). The server machine then performs any required processing and responds to the client.

SBC presents many advantages over traditional client-based computing. Hardware and software updates, application deployment and product support can easily be managed by publishers on the server side. Server-based architecture is by nature a centralized technology model, thereby increasing accessibility for those managing the software and allowing quicker and more thorough mitigation of potential security risks. The centralized nature of SBC also creates a more scalable model. Increased usage is accommodated by adding new servers for distributed handling of a larger number of clients (Howorth 2006).

Users expect to be able to use applications across a variety of platforms, each of which will possess limiting factors for application deployment such as hardware constraints, varying bandwidth, different operating systems and potential software conflicts. Sufficient performance, security and OS-independence are more predictable and therefore more easily achieved via server-based computing. The limited requirements placed upon the client-side in a server-client relationship make the client effectively just a service terminal. Clients that perform the majority of their services via servers may often be replaced by less expensive and more easily managed units called "thin clients". (Howorth 2006)

2.1.1 Thin Clients

Thin clients rely upon servers to perform the majority of an application or service's operations. Basic input and output devices such as a keyboard, monitor, mouse and printer are provided as well as the required memory and processing power needed to sufficiently interact with the server. There's typically less emphasis on local data storage though temporary data-caching may be used for better efficiency. The minimal requirements on

client side increase the average life of the hardware used (6 years on average vs. 3) (Howorth 2006). Expenses related to the management and support of these clients is also greatly reduced.

2.1.2 Thick (Fat) Clients

By contrast, thick (or “fat”) client applications are decentralized in terms of processing power. Processing by the server is relatively unsophisticated with the majority of the work conducted on client-side. Generally thick-client applications have the following characteristics (McKenna 2002):

1. Greater bandwidth requirements on a WAN
2. Greater PC maintenance costs
3. Less application control (harder to push updates)
4. Greater cost for client PC maintenance

2.2 GIS Servers

GIS servers allow for the posting of custom maps to the internet for interaction with a broad audience. There are several GIS server options available for implementation including both proprietary and open-source options. There are several considerations when weighing the benefits of leveraging one of several Enterprise solutions vs. their open-source counterparts. Some of the benefits of open-source GIS would include:

- Inexpensive or free base package
- Potentially enthusiastic community support (particularly noticeable with GeoTools highlighted in Section 2.2.1)
- Flexible custom development options (source code available for contribution, alteration and configuration)

Some of the costs would include:

- Support limited to forums, volunteer community support
- Potential for halted development of the product

The alternative, Enterprise software can include the following benefits:

- Extensive support options (community-based, live tech-support, dedicated account handler, etc.)
- Industry-leading technology
- Widespread adoption

However some of the costs would include:

- High entry cost
- Limited access to code base for custom configuration

As of the final writing of this paper (November, 2013), the University of North Carolina Wilmington had already purchased a software package site license that includes Esri ArcGIS Server 10.1. Implementation of this package represented a practical use of previously purchased software that was in-line with the Department of Geography and Geology's instructional objectives. It was therefore unlikely that an alternative GIS server solution would be implemented in this particular case. It was however worth investigating what options were available, should alternative solutions be considered at a later time.

2.2.1 Open-Source and/or Free GIS Server Solutions

Given the extensive need for mapping capabilities worldwide, there are several open-source desktop GIS solutions available (Wikipedia 2013). For publishing of map data developed on these desktop applications as services, there are a number of open-source GIS Server options (as of 2012), including the following (GISLounge 2012):

1. GeoTools: A Java-based GIS toolkit containing several GIS utilities including GeoServer. It possesses a modular architecture making it simple to add extra functionality.
2. GLG Map Server: The GLG Community Edition Map Server is an OpenGIS-compliant map server. The map server supports CGI and standalone map interfaces.
3. KIDS (Key Indicator Database Systems): A GIS and thematic mapping application developed by a sub-organization of the United Nations. Originally developed to map food-scarcity regions it's now capable of posting a broad array of statistical map data. The application is primarily Java and XSL document-based.

4. Map Server: A platform for publishing interactive mapping applications. Cross-platform functionality that supports development in multiple languages.
5. Map Zoom: A basic application for posting clickable, zoomable maps on the internet.

This is by no means an exhaustive list of available options however any of these options could theoretically be implemented to achieve similar results with little to no monetary investment (just the use of current hardware resources).

2.2.2 Enterprise GIS Solutions

There are several other companies offering GIS solutions having similar capabilities to Esri's ArcGIS software, though often times competitors offerings focus on a specific market niche.

- MapInfo: A software product that offers map data solutions for GIS business applications (MapInfo 2013)
- Intergraph: Provides geospatially-driven solutions for a variety of industries including government, public safety and utilities (Intergraph 2013)
- AutoDesk: Offers AutoCAD 3D software with GIS capabilities suitable for a variety of industries (Autodesk 2013)
- GE Energy: Offer the SmallWorld GIS software, primarily targeted for use by the utilities industry

Each of these companies offer well developed and presumably similarly well-supported competing products to ArcGIS's software, though each appeared more directly targeted at specific industries and for specific applications when compared to Esri's ArcGIS product. Though perhaps not as appropriate for general instruction, for specific projects, each of these could be very well suited for the task.

2.3 Analysis & Design

There are a number of different approaches for the analysis & design of a system. Choosing which approach is most suitable for a particular situation required assessment of the particular requirements of the customer as well as the requirements and limitations of the system.

2.3.1 Object-Oriented Analysis and Design

Object-oriented analysis and design (OOAD) is “A software engineering approach that models a system as a group of interacting objects. Each object represents some entity of interest in the system being modeled, and is characterized by its class, its state (data elements), and its behavior.” (“Object-oriented analysis and design” 2013)

A system operates in an environment composed of people and things; each component of this system interacts with other components to create some desired result. OOAD allows a user to view system requirements from the perspective of classes and objects. From the code-level, a person can then focus on what that system is supposed to do. People and things in the system are abstracted to create classes. Individual characteristics and behaviors can be analyzed to determine the different states and functions an object should have. This analysis used to identify each object and its actions aides in the design and development of the overall system. (Sauter 1999)

2.3.2 Structured Analysis and Design Technique

Structured Analysis and Design Technique (SADT) is “A systems engineering and software engineering methodology for describing systems as a hierarchy of functions.” (“Structured Analysis and Design Technique” 2013)

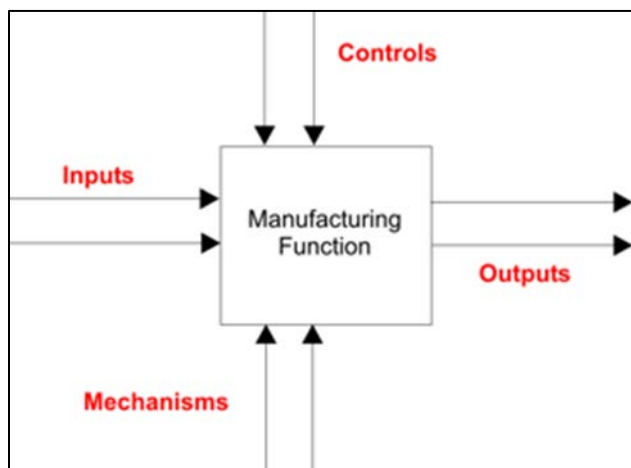


Figure 1: Example of SADT ("IDEF0.", n.d.)

SADT uses diagrammatic notation for composing a visual representation of an application. Entities and activities are represented using boxes; these are then related to each other using different styles of arrow, often labeled to communicate data flow or relationship.

2.3.3 Service-Oriented Analysis and Design

Service-oriented analysis and design (SOAD) is “The discipline of modeling business and software systems, for the purpose of designing and specifying service-oriented business systems within a variety of architectural styles, such as enterprise architecture, application architecture, service-oriented architecture, and cloud computing.” ("Service-oriented modeling" 2013)

According to Zimmerman, Krogdahl and Gee (2004), there are three levels of abstraction within Software Oriented Architecture:

- Operations: Transactions of single logical units of work. These transactions typically involve read, write or modification of data. Execution of one operation may launch additional operations
- Services: Logically associated operations.
- Business processes: A series of actions or activities performed with the intent of accomplishing specific business goals. These processes usually consist of a series of operations which are executed in an ordered sequence.

2.4 ArcGIS Performance and Scalability

2.3.1 GIS Server Terminology

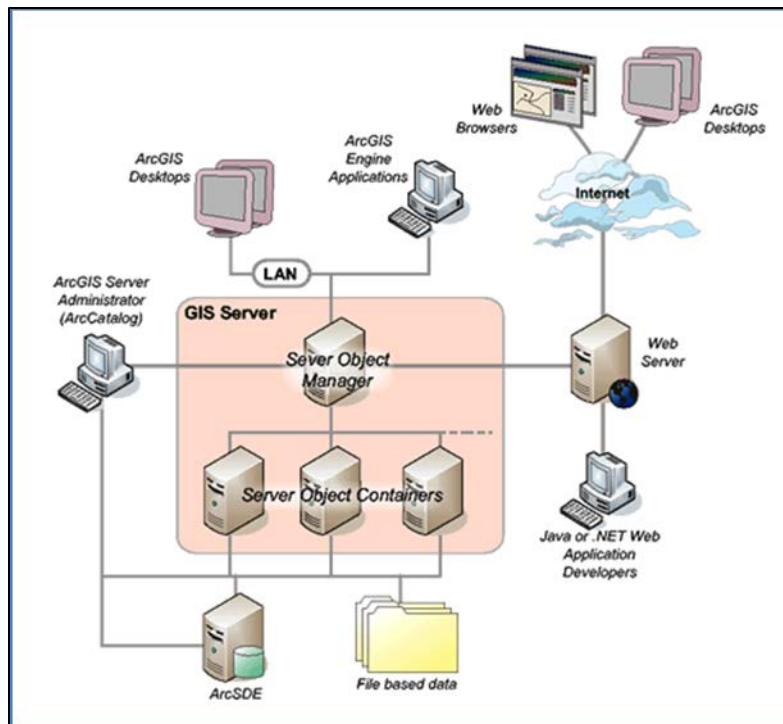


Figure 2: An example of an ArcGIS Server 9.2 configuration. (Esri 2006)

There are several unique functional components of a GIS server system. Many of these components will be similar across different GIS servers however the terminology presented below are specific to ArcGIS Server versions 9.2, 10.0 and 10.1.

- **ArcGIS Server Administrator:** The individual(s) responsible for administration of the ArcGIS Server. (Esri 2006)
- **ArcSDE:** The Arc Spatial Database Engine enables relational database management systems for interaction with spatial data. (Esri 2006)
- **File based data:** A dataset existing as a single file or group of related files. (Esri 2006)
- **Server Object Manager (SOM):** A service, ideally running on a single machine that manages a collection of Server Object Containers. (Esri 2006)
- **Server Object Containers (SOC):** Can be installed on one or several machines. SOC machines host the services run by the ArcGIS Server. Server Object Containers can only have one SOM. (Esri 2006)

- ArcGIS Engine Applications: Standalone applications employing ArcGIS functionality. (Esri 2006)
- Web Server: A computer that delivers web content to a requesting machine. (Esri 2006)
- ArcGIS Desktop: A machine running ArcGIS desktop software used for creating an ArcGIS Server map service. (Esri 2006)
- Site: A site is composed of several components such as a GIS server and the Web Adaptor. (Esri 2012)
- GIS server: Maintains service operation and configuration, handles and responds to requests issued to the GIS web services. A GIS server draws maps, run required tools, serves imagery, and executes numerous other operations offered by ArcGIS. (Esri 2012)
- ArcGIS Web Adaptor: This optional component allows you to configure a web entry point into your site. It integrates with your web server and then passes incoming requests to connected GIS servers. (Esri 2012)
- Server directories: Directories containing files that support your service; includes cache, search indexes, and geoprocessing job results. (Esri 2012)
- Configuration store: A directory containing configuration information such as the list of GIS servers participating in the site. (Esri 2012)
- Data: Data that supporting web services; may include feature classes, tools, imagery, and locators. (Esri 2012)

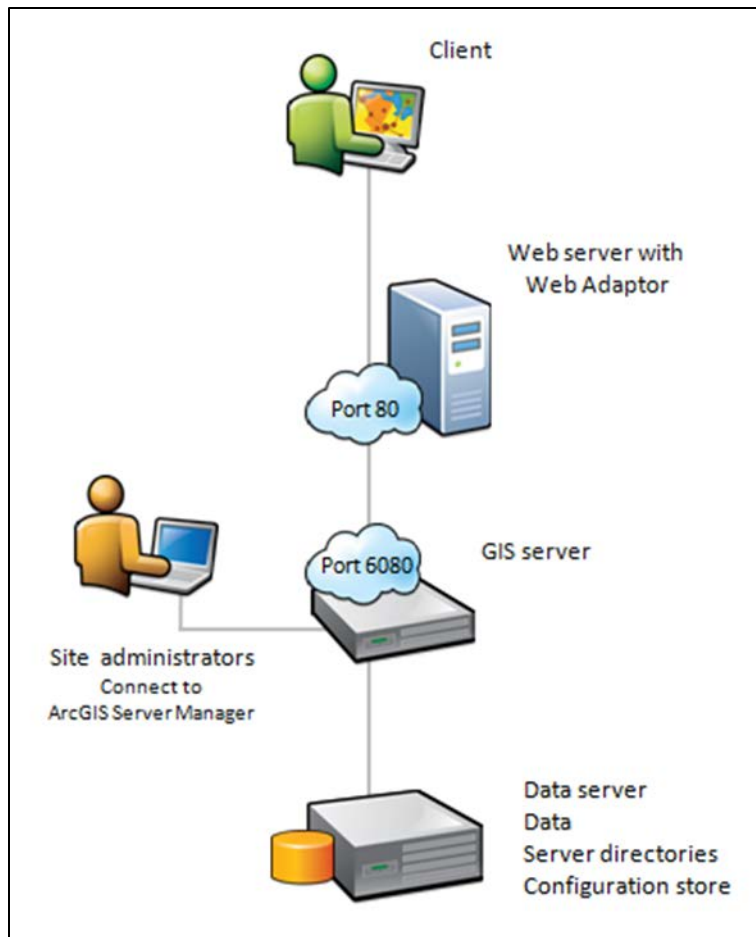


Figure 3: An example of an ArcGIS Server 10.1 configuration (Esri 2012)

2.3.2 GIS Server Performance and Scalability

There's tremendous room for variation in the configuration of an ArcGIS Server installation depending on the requirements and resources of the user(s). The server's components can all be run on a single machine or they can be distributed across several machines. Considerations for deployment configurations might include the number of services that will be simultaneously deployed, the data-intensive nature of the applications, how computationally-intensive the application geoprocessing tasks will be and how many users will be accessing the services at any given point in time. A few common configurations include:

- Single machine (physical or virtual): It's possible to install the ArcGIS Server and Desktop on the same machine as the webserver. For this setup to work, the machine must satisfy the requirements of both the GIS Server and the chosen Application Development Framework used to develop services on the desktop application. (Esri 2006)
- Multiple ArcGIS Server machines: As the processing load on the ArcGIS Server increases, one option is to increase the number of Server Object Containers (on UNCW's version 9.2 and 10.0 servers) or the number of ArcGIS Servers (on UNCW's version 10.1 and 10.2) across which services are published. For ArcGIS Servers prior to version 10.1, it's not required that SOC's be installed on the same machine as the SOM. (Esri 2006)
- Dedicated Web Server: ArcGIS server components can be installed on one machine or several while the web server can be installed on a separate machine. In this configuration, the Application Development Framework must be installed on the web server. (Esri 2006)

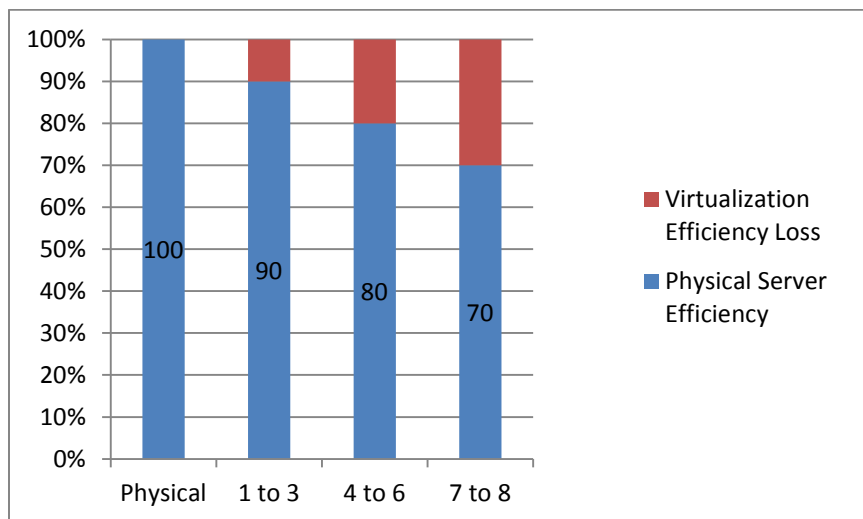


Figure 4: Data tested in an Esri lab, presented by Sakowicz and Pizzi (2012) outlining efficiency losses on virtualized CPUs.

The “limits” of a specific ArcGIS Server installation fall more under the category of “general performance & scalability”. Esri (Sakowicz & Pizzi 2012) defined these terms at an Esri Developer Summit as follows:

1. Performance: The speed at which a given operation occurs
2. Scalability: The ability to maintain performance as load increases

Both performance and scalability may be monitored using third party software or simply through user observation (website loading delays). One dilemma that may be encountered when pushing the limits of the current configuration is whether it's best to upgrade the VMware image hardware or to install more instances of ArcGIS Server. Sacowicz and Pizzi (2012) tested machines at Esri's Labs with the configurations listed below in Figure 5:

# and type of machine(s)	# Sockets	# Cores/Socket	RAM/VM	Total # Cores across machines	Total RAM across machines
1 Physical	2	4	32 GB	8	32 GB
1 Virtual	2	4	32 GB	8	32 GB
2 Virtual	1	4	16 GB	8	32 GB
4 Virtual	1	2	8 GB	8	32 GB
8 Virtual	1	1	4 GB	8	32 GB

Figure 5: Configuration of VMs running ArcGIS Server for performance testing by Bhattacharjee et al. (2013)

The team reported <1.26% difference between the maximum throughput of the physical machine and the VM configured with 8 cores. The VM configuration of 1 VM with 8 CPUs was expected to perform better than the other VM configurations as mentioned by Bhattacharjee et al. (2013), "This is expected behavior, as ArcGIS for Server generally performs better when CPUs are distributed across fewer machines." Similar performance comparisons were made for response time with the physical and virtual machines containing 8 CPUs responding at similar speeds, faster than the other VM configurations.

The greatest efficiency was achieved with a physical machine however this may be more difficult to provision, maintain and upgrade. Using virtual machines, there's a progressive decline in efficiency as the number of CPUs is increased. The most likely scenario is to find the best combination of administration, efficiency and relative cost as determined by the primary administrators: Dr. Halls and UNCW ITSD. A few scenarios prioritizing each of these categories are presented later in Section 4.

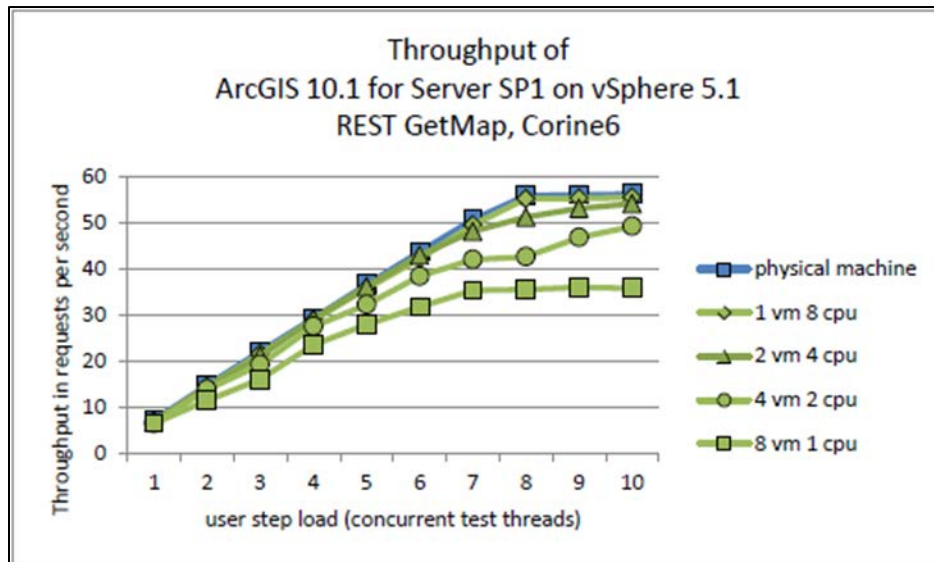


Figure 6: Results obtained by Bhattacharjee et al. (2013) when testing ArcGIS Server 10.1 as installed on a physical machine against multiple VM configurations.

Further testing of ArcGIS Server 10.1 was performed jointly by VMware and Esri to characterize ArcGIS Server 10.1's performance on the VMware vSphere® 5.1 virtualization infrastructure (Bhattacharjee, Baleja, Perez, Catanzano 2013). The team from Esri and VMware tested the configurations listed in Figure 6. Each of the configurations was tested both in terms of its performance relative to the other VM configurations as well as its performance on the machine vs. a comparable physical machine's maximum throughput.

Bhattacharjee et al. (2013) applied the test load in steps starting from 1 thread and ending at 10 threads using the CORINE European program dataset (publicly available from the European Environmental Agency).

2.5 Web Portal Design Specific to UNCW

The back-end details of the ArcGIS Server are by nature significantly more complex than the front-end. The front-end of the server is accessible through the web portal which is located within the domain: <http://www.uncw.edu>. This domain is operated by the University of North Carolina Wilmington. Individual departments within the University are given permissions to modify content specific to their departments within the University's preestablished guidelines. The department's GIS portal is located here: <http://www.uncw.edu/gis/webgisportal.htm>.

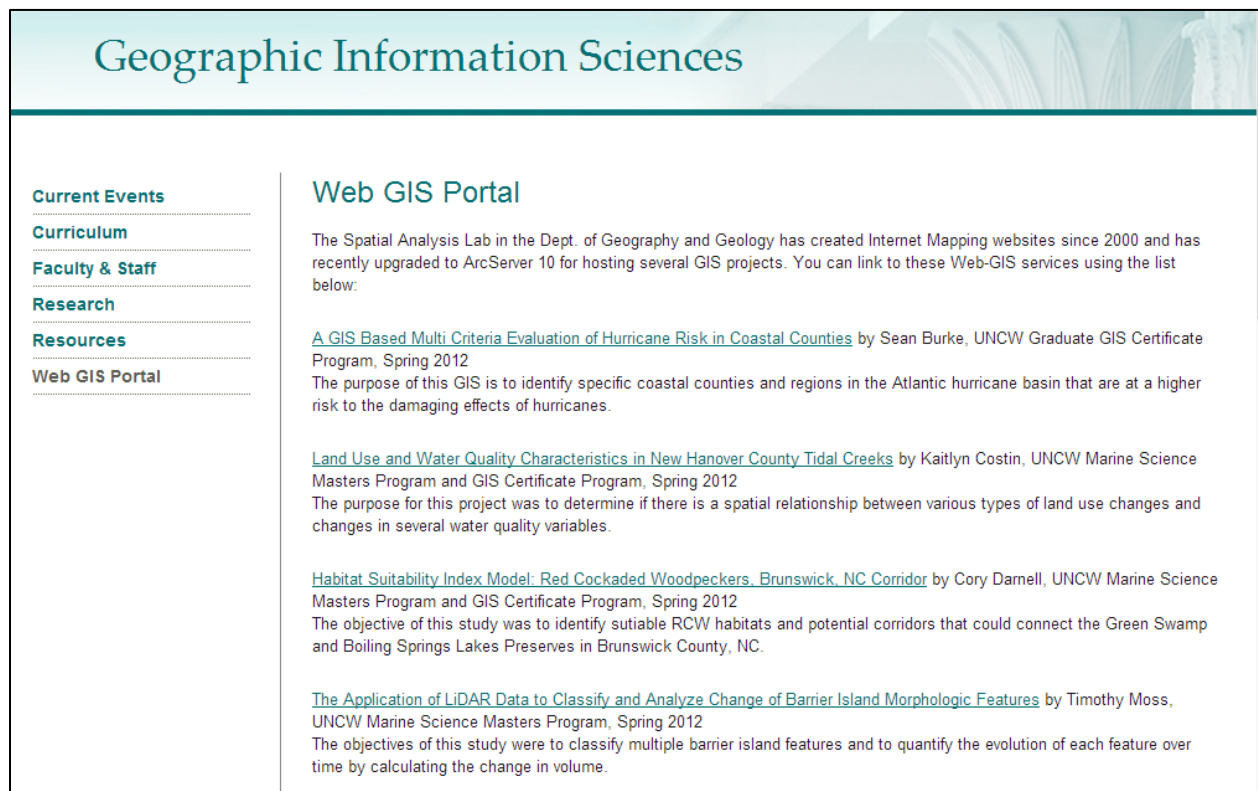


Figure 7: Current UNCW GIS portal.

Making the required alterations to the portal requires a basic understanding of web design using HTML and Javascript. Given the basic web update requirements, a thorough literature review on web design was determined to be out of scope for this portion of the project.

The web portal contains links to several UNCW student projects created in Spring, 2012. Projects created prior to this time are currently linked on the site however they reside

on the previous server. Once projects have been migrated to the new server, the portal's links and project descriptions would be updated to reflect the newly accessible projects on the new server. Tests of the hyperlinks in the portal indicate that roughly half of the currently linked applications are working.

3. PROJECT METHODOLOGY

3.1 Design of Research

3.1.1 Tools & Applications

Each phase of this project employed methodologies requiring unique functional utilities and concepts, some would be used across multiple project phases. The tools and applications used are listed at the beginning of each sub-section under 'Project Execution' and are also all listed together in a chart in Appendix T.

3.1.2 Timeline

The final revised project final date for project completion and defense is December 02, 2013. Figure 9 summarizes the milestones to be achieved prior to project completion:

Milestone	Date Estimate	Deliverable	Confidence Level
Project Definition	September 28, 2012	Define project goals Determine configuration options Estimate required resources	Medium
Analysis & Design of Web Server	February 20, 2013	Actor diagrams Class, Sequence, Activity and State Diagrams Software Requirements Specification (SRS)	Medium
Web application migration	November 10, 2013	Oculina and Riverrun web applications confirmed to function on new server	Medium
Sample application creation	November 11, 2013	Sample application template for student use created and posted to GIS Portal	High
Web portal update	November 15, 2013	Links/descriptions of web applications posted to UNCW's GIS portal	Medium
Project completion	December 02, 2013	Determine that project meets stakeholder requirements within given constraints Project defense	Medium

Figure 8: Project timeline. Can also be found in the context of the project charter in Appendix B, Section 4.5

Further details for each deliverable are given below in 'Outcome Measures'.

3.1.3 Outcome Measures

The project's status can be measured in terms of completeness given its position relative to the following milestones and deliverables, these outcome measures are written in the future tense as they were defined during project inception:

Milestone	Deliverable
Web application migration	Oculina and Riverrun confirmed to function on the new server with a similar level of functionality as permitted in the new environment
Sample application template	A sample web application from one of Esri's JavaScript templates will be modified to allow "novice developer" code modification by UNCW GIS students for application creation. This will be accompanied by instructions for proper application creation.
Web portal update	<p>Completion of the GIS portal update will be met by completion of the following two requirements:</p> <ol style="list-style-type: none"> 1. (pre-requisite to website modification) Web Content Manager and Editor Training completed. This training is mandated by UNCW for staff, faculty and students who wish to modify websites representing either the university or one of its constituent colleges 2. Successful application migration and subsequent posting of publicly accessible URLs with accompanying descriptions. 3. Link for sample template
Installation of ArcGIS 10.1, Analysis & Design of System	<p>ArcGIS 10.1 shall be installed to perform at the same level of functionality as that of the 10.0 server system.</p> <p>For the analysis & design, the following items shall be delivered for a complete conceptual view of the web server:</p> <ul style="list-style-type: none"> • Actor diagrams • Instructions for use (Use cases) • Business process models for service and application creation on each of the three servers • System component analysis, wireframes for application design

Figure 9: Project timeline, can also be found in the context of the project charter in Appendix B, Section 4.4.

Project completion relative to these deliverables is assessed in Section 4.

Pertaining to the outcome measures, it should also be stated that the scope of the project evolved in many ways to accommodate the original project's intent. Overall changes to the project may be seen by comparing the original proposed Project Charter in Appendix A with the final Project Charter in Appendix B. This is discussed further in Sections 4 and 5.

3.2 Project Execution

3.2.1 System Analysis

3.2.1.1 Analysis & Design Tools & Methodology

For the Analysis & Design, the following tools were used:

- Microsoft© Windows™ 7 64-Bit OS
- Google© Chrome™ Version 29.0.x
- Cisco© AnyConnect™ VPN Client will be used for remote access to the UNCW network
- ESRI© ArcGIS Server™ Enterprise Edition 10.1
- ESRI© ArcGIS Server™ Enterprise Edition 10.0
- ESRI© ArcGIS Server™ Enterprise Edition 9.2
- CrystalDiskInfo 5.6.2

The Analysis & Design of UNCW's ArcGIS Server system was split such that the Analysis is covered here in Section 3 and the Design (drawn from the results of the analysis) is covered as part of the results in Section 4. For the Analysis & Design, a Service Oriented Analysis & Design (SOAD) (Section 2.3.3) approach was employed. Other methodologies were considered as outlined in section 2.3. Object-Oriented Analysis & Design (Section 2.3.1) focuses primarily on application-level functionality. Structured Analysis & Design Technique (Section 2.3.2) represents a higher level approach than OOAD but fails to capture business processes. With its focus on operations, services and business processes, SOAD captures an appropriate level of detail for a mixed audience of IT professionals, software developers and both technical and non-technical UNCW faculty and staff. It's also a flexible analysis technique, allowing us to pull from aspects typical of other

Analysis & Design methodologies. Adapted from the elements of SOAD listed by Zimmerman et al. (2004), the following are present in the Analysis & Design:

- Process and notation in the form of Business Process Models Notation (BPMN)
- Structural conceptualization:
 - Modeling of application and services
 - Modeling of system components
- Quality factors and best practices
- Be able to answer the question: What is not a good service? For example, how reusable is your service?
- End-to-end modeling and business, architecture and application design.

The methods employed to meet these criteria for the System Design are addressed in Section 4.

To perform an analysis of the systems, it was necessary to first determine how the GIS Servers systems had been configured, how they were similar to each other and how they differed. There are currently 3 VMware servers hosting ArcGIS servers. The following ArcGIS Server versions are represented:

- ArcGIS Server 9.2
- ArcGIS Server 10.0
- ArcGIS Server 10.1

3.2.1.2 ArcGIS Server 9.2 Overview

3.2.1.2.1 System

The ArcGIS 9.2 Server is hosted on a VMware machine with the following configuration:

Configuration Item	Description
Computer Model	ACPI Multiprocessor PC
Processor	Intel® Xeon® CPU E5640 @ 2.67 GHz
Memory	2.66 GHz, 2.00 GB of RAM
Hard drive capacity	49.8 GB
Network card	VMware Accelerated AMD PCNet Adapter
# Server instances on machine	1
Operating system	Microsoft Windows Server 2003, Standard Edition, Service Pack 2
Computer/Network name	//#####.das.uncw.edu

Figure 10: Configuration of server hosting ArcGIS Server 9.2

This VMware image is part of a larger system designed for GIS service and application creation and deployment. Components of this system are illustrated in Figure 11.

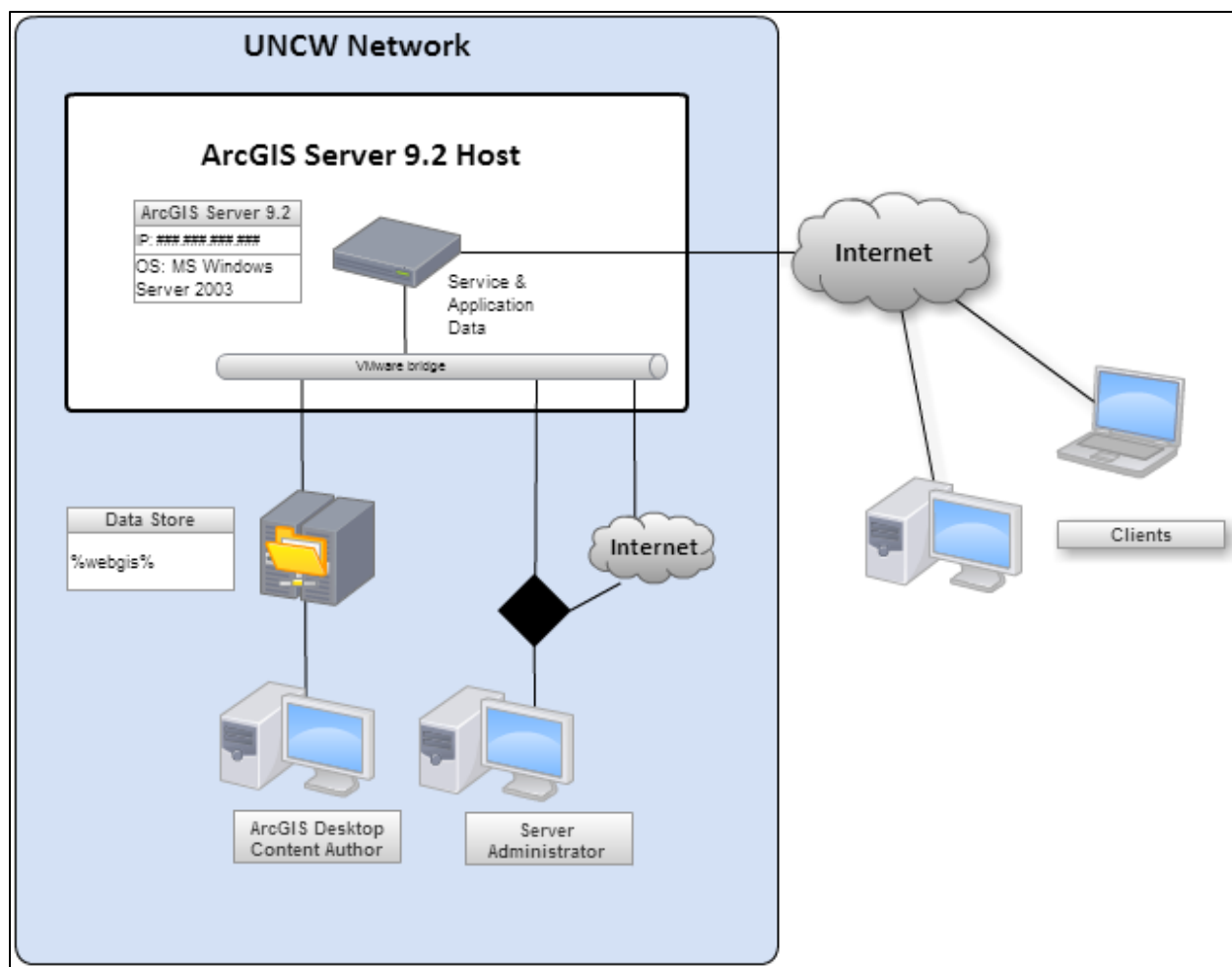


Figure 11: System components employed for service and application creation for UNCW ArcGIS 9.2 Server. A larger version of this diagram may be found in Appendix I.

The VMware guest can be further divided into the sub-components shown in Figure 12. ArcGIS Server 9.2 uses its own web server which could be deployed separately, but in this case was deployed on the same machine as ArcGIS Server. Two ArcGIS server sub-components, the Server Object Manager (SOM) and Server Object Container (SOC), are of course, also present on the VM.

3.2.1.2.2 Configuration

ArcGIS Server 9.2 consists of multiple components required for successful service and application deployment. The UNCW ArcGIS 9.2 Server setup was observed to have the

following components installed on the same VMware machine image (For component reference see Figure 5):

- Server Object Manager (SOM)
- Server Object Container (SOC)
- Web server

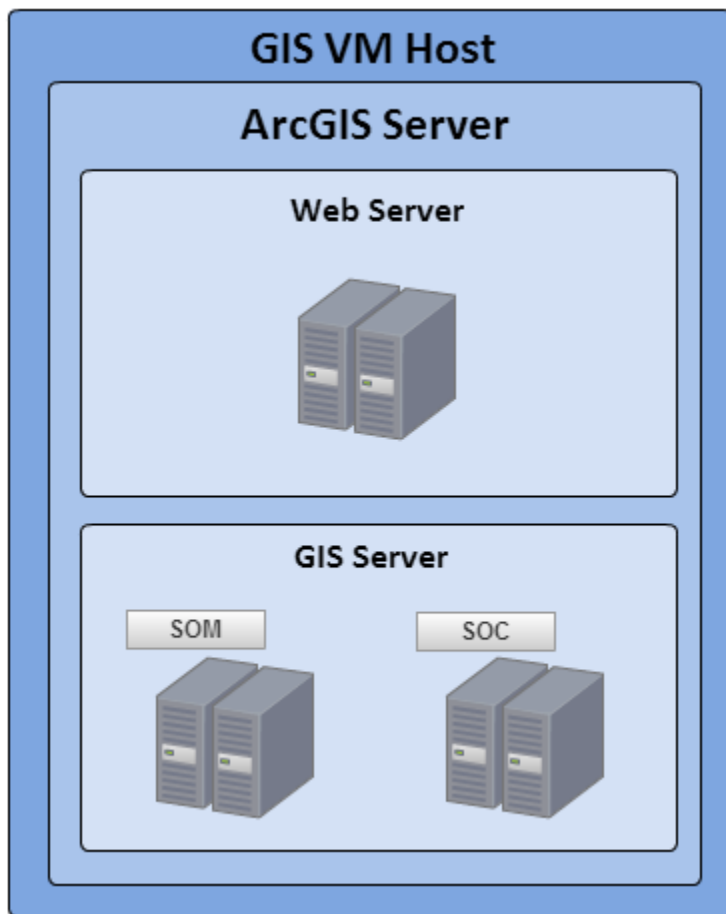


Figure 12: ArcGIS 9.2 GIS VM server configuration (can also be found in Appendix I).

The ArcGIS Server was understood by the department to have access to a network folder (data store) and is modeled as such in Figure 11. As of the date of observations (September 25, 2013), only a local directory was listed in the server's configuration manifest. ArcGIS Server 9.2 does allow for multiple data stores, therefore it is possible there may previously have been more. It was also observed on this date that the domain account that had been

created for use by the SOC had been given explicit read/write permissions to the network folder implying it may once have been connected though it no longer appears to be.

3.2.1.2.3 Process for Service & Application Deployment

A business process model for this process can be found in Appendix L. ArcGIS Server 9.2 users at UNCW used the following process for service and application creation (See Figure 11 for system reference):

1. User authors map(s) using ArcGIS Desktop 9.2
2. User with administrative privileges logs into ArcGIS Server 9.2
3. User copies map data to shared data store (if data not already there) and then publishes map as service using ArcGIS Desktop-Server interface
4. User with VMware //##### access privileges logs into machine instance
5. User with ArcGIS Sercer 9.2 administrative privileges creates application to consume service created in previous steps using server-provided application tool/viewer
6. User publishes finished application using ArcGIS 9.2 web server to public facing URL: http://#####.uncw.edu/<application_name>/viewer.htm where “<application_name>” is the user-provided name for the application

3.2.1.3 ArcGIS Server 10.0 Overview

3.2.1.3.1 System

The ArcGIS 10.0 Server is hosted on a VMware machine with the following configuration:

Configuration Item	Description
Computer Model	ACPI Multiprocessor x64-based PC
Processor	Intel® Xeon® CPU E5640 @ 2.67 GHz
Memory	2.66 GHz, 3.00 GB of RAM
Hard drive capacity	39.9 GB
Network card	Intel® PRO/1000 MT Network Connection
# Server instances on machine	1
Operating system	Microsoft Windows Server 2003 R2, Standard Edition, Service Pack 2
Computer/Network name	////###.###.###.###

Figure 13: Configuration of server hosting ArcGIS Server 10.0

Components of the system used for service and application creation are illustrated in Figure 14.

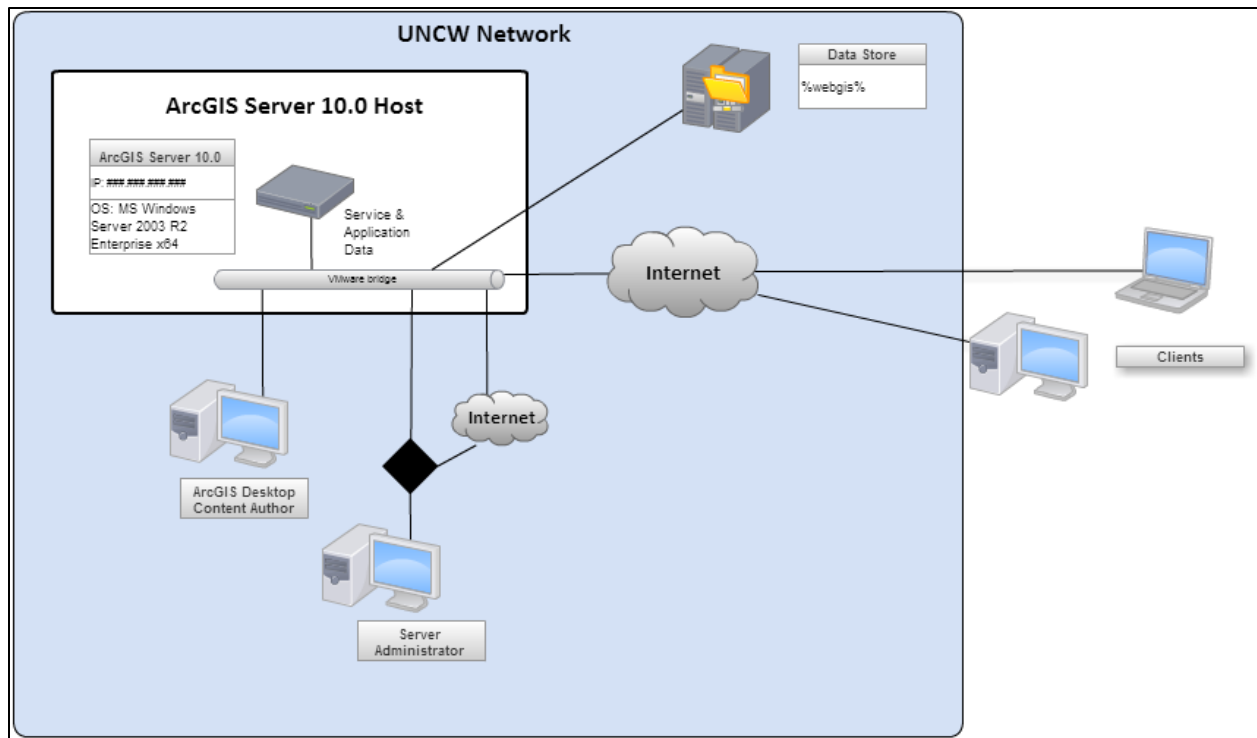


Figure 14: System components employed for service and application creation for UNCW ArcGIS 10.0 Server. A larger version of this diagram may be found in Appendix J.

The WebGIS portion (shown in Figure 14 as VMware guest) can be further divided into the sub-components shown in Figure 15. Like ArcGIS Server 9.2, ArcGIS Server 10.0 uses its own web server which could be deployed separately, but in this case was deployed on one machine. Also like ArcGIS Server 9.2, the two GIS server sub-components, the SOM and SOC, are also installed on the same virtual machine.

3.2.1.3.2 Configuration

ArcGIS Server 10.0 consists of multiple components required for successful service publishing and application deployment. The UNCW ArcGIS 10.0 Server setup was observed to have the following components installed on the same VMware machine image (Figure 15):

- Server Object Manager (SOM)
- Server Object Container (SOC)
- Web server

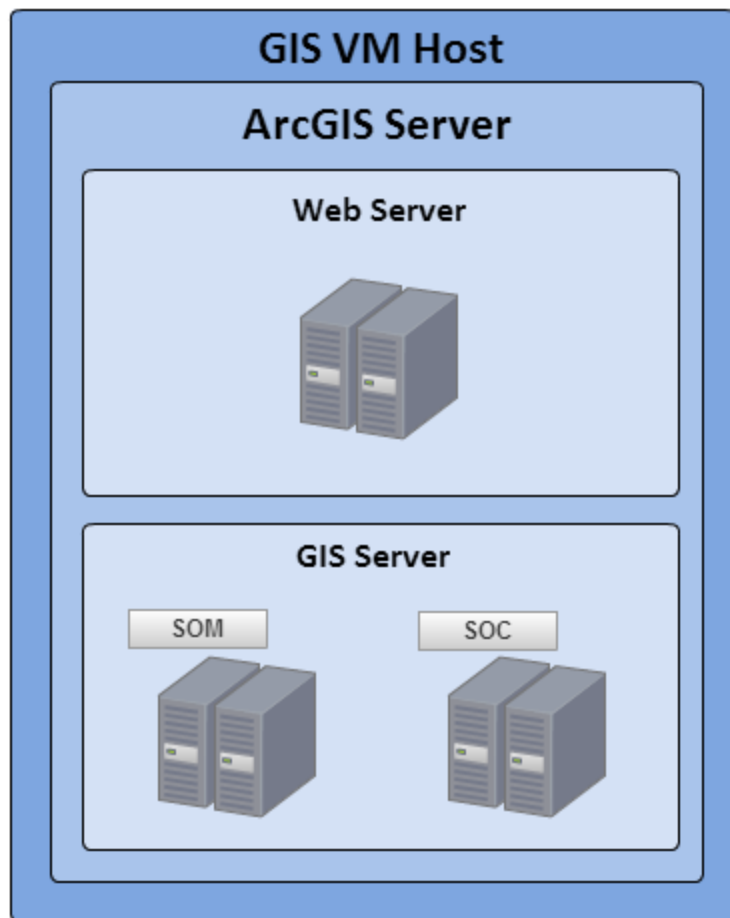


Figure 15: ArcGIS 10.0 GIS VM server configuration (can also be found in Appendix J).

3.2.1.3.3 Process for Service & Application Deployment

A business process model for this process can be found in Appendix M. ArcGIS Server 10.0 users at UNCW used the following process for service and application creation (For component reference see Figure 14):

1. User creates map(s) on ArcGIS Desktop 10.0
2. (Optionally) User with publisher privileges publishes map to ArcGIS Server 10.0
3. (Optionally) User creates service using ArcGIS Server interface and uploads map data to data store located on
4. User with VMware administrative privileges logs into server01 VMware machine instance

5. User creates application using server provided interface
 - a. If steps 2 and 3 completed: application consumes service, else: user selects layers for application from data store
6. User publishes finished application using ArcGIS 10.0 web server to public facing URL: `http:// ###.###.###.###:8399/<application_name>/mapviewer.jsf`, where “<application_name>” is the user-provided name for the application

3.2.1.4 ArcGIS Server 10.1 Overview

3.2.1.4.1 System

The original intent of this project didn't include setting up ArcGIS 10.1 server. This task arose when it was later determined that the upgraded desktop software (ArcGIS 10.1 for Desktop) was not capable of publishing services to the ArcGIS 10.0 server. Given the choice between migration to an obsolete server and the installation of a new server, Dr. Halls and I reached an agreement for me to set up an instance of the 10.1 server; the following conditions of the agreement were made:

1. A member of ITSD would be in charge of provisioning a VMware server instance with similar capabilities (per ITSD discretion) to the one used to host ArcGIS Server 10.0
2. Once the VMware server was up and running, I would configure the ArcGIS 10.1 server to a similar level of capability as that demonstrated by the 10.0 server per my discretion
3. The applications requiring migration would be migrated from the version 9.2 server to the version 10.1 server
4. The Analysis & Design would be performed on the 10.1 server

The ArcGIS 10.1 Server would be hosted on a VMware virtual machine with the following configuration:

Configuration Item	Description
Computer Model	ACPI Multiprocessor x64-based PC
Processor	Intel® Xeon® CPU E5640 @ 2.67 GHz (2 Processors)
Memory	2.66 GHz, 4.00 GB of RAM
Hard drive capacity	39.8 GB
Network card	Intel® PRO/1000 MT Network Connection
# Server instances on machine	1
Operating system	Microsoft Windows Server 2003 R2, Standard Edition, Service Pack 2 (64-Bit)
Computer/Network name	//###.###.###.###

Figure 16: Configuration of server hosting ArcGIS Server 10.1.

3.2.1.4.2 Installation

There were several options available for configuration of the ArcGIS Server 10.1 installation. In each instance, software configuration was performed to the standards required to best reflect the configuration of the previous 10.0 Server. The following items were noted during installation:

1. Installation was performed for a default, single-machine configuration.
2. 32-bit systems are no longer supported (not an issue for this installation).
3. ArcGIS Server 10.1 communicates through port 6080 for http traffic, this is currently only used by the web adaptor which uses port 80 and will allow viewing of services using a URL that doesn't require specification of port #
(http://gis.uncw.edu/<application_name>/)
4. An authorization file was required to complete the installation; this file is provided by Esri. In UNCW's case, a new authorization file is required every year to renew the education license.

5. The web adaptor was not required for installation but would be required (in conjunction with a separate web server) to host web applications and present them to the public without a port # in the URL. The adaptor was later installed on a standalone, publicly-accessible server (#####.uncw.edu).
6. ArcGIS Server requires an account with privileges to read/write privileges for folders used as data stores. This account can be either a local account or a domain account. Domain accounts are more flexible as they allow mapping of network drives/folders for data stores. I used an account that was set up with the correct permissions for the ArcGIS Server 9.2 installation:
 - a. Username: uncw\#####
 - b. Password: #####

The WebGIS portion (shown in Figure 17 as VMware guest) can be further divided into the sub-components shown in Figure 18. ArcGIS Server 10.1 is capable of hosting services on a local web server similar to previous versions however, unlike its predecessors, this server can't host applications. The server setup was initially configured to host applications one of two ways: using ArcGIS Online or they could be deployed locally using an installed instance of Apache Tomcat 7.0.42. Also unlike the previous two ArcGIS servers, as of the 10.1 release the SOC and SOM components are no configured by the user. Instead, multiple ArcGIS 10.1 Servers may be set up and configured to work together to distribute the workload. This step was unnecessary in this case as it was a single-machine deployment.

The Apache-Tomcat server method for application deployment was later overhauled as another VM with IIS 7 was added specifically for the ArcGIS Web Adaptor and application deployment. Rather than deploying applications locally on the same machine as ArcGIS Server, applications could instead be deployed on `http://###.###.###.###` and would be visible to the outside world.

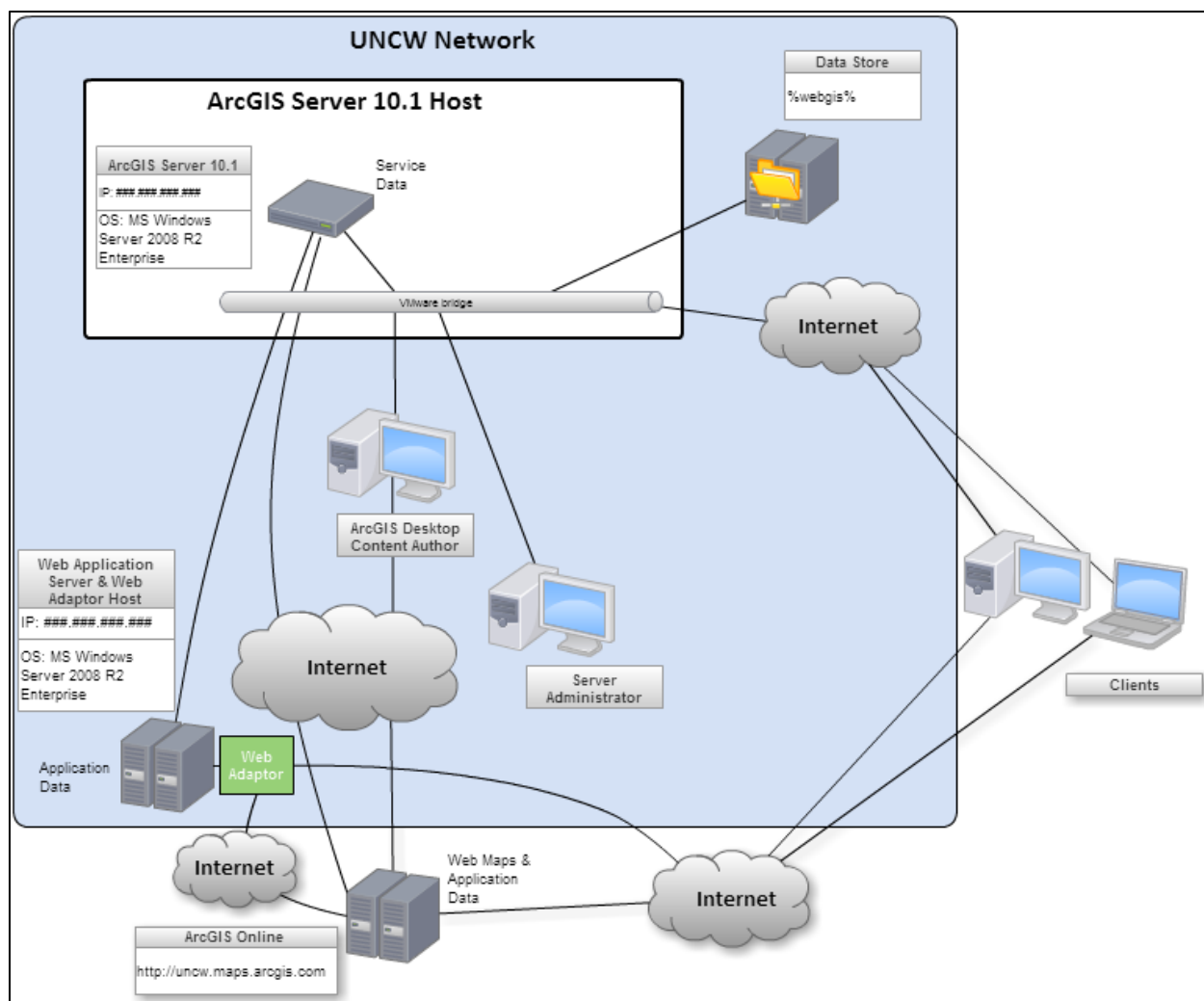


Figure 17: ArcGIS Server final system configuration for service and application creation. A larger version of this diagram may be found in Appendix K.

3.2.1.4.3 Configuration

ArcGIS Server 10.1 consists of multiple components required for successful service and application deployment. The UNCW ArcGIS 10.1 Server setup was configured similarly to the 10.0 installation to have the following components installed on the same VMware machine (Figure 18):

- Application Web Server (also host for ArcGIS Web Adaptor)
- GIS Server
- Data store

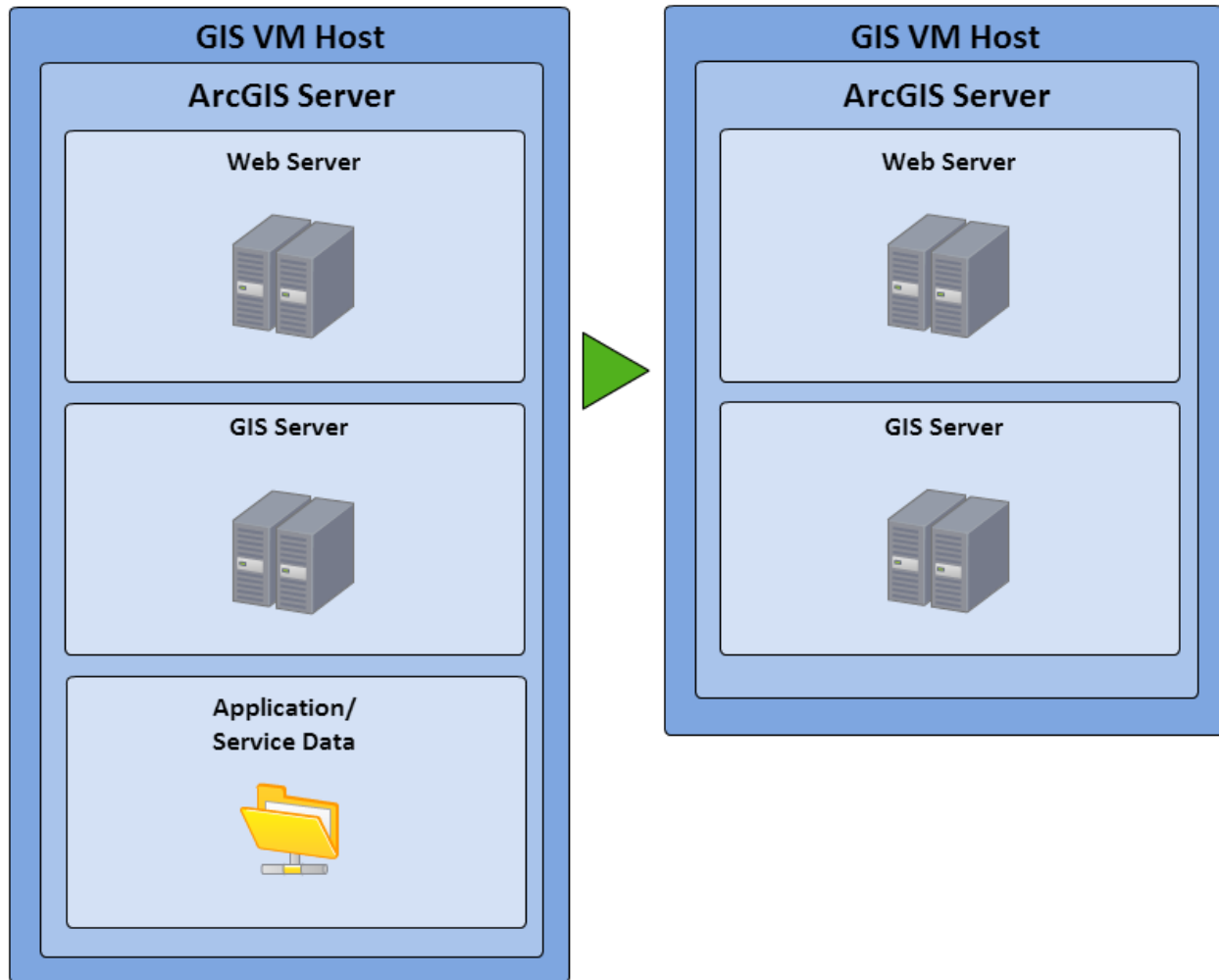


Figure 18: ArcGIS 10.1 GIS VM server configuration began including the Application data but later was configured to pull service data from the preferred source of the Network Attached Storage (NAS) //arcshare (can also be found in Appendix K).

3.2.1.4.4 Process for Service & Application Deployment

A business process model for this process can be found in Appendix N. ArcGIS Server 10.1 users at UNCW use the following process for service creation followed by either the basic or the advanced process for application creation (For component reference see Figure 17, more detailed documents covering service and application creation may be found in Appendices P, Q, R & S):

Service creation:

1. Create map(s) on ArcGIS Desktop 10.1, if map data is not already in the designated webgis folder, create a unique folder in webgis and transfer data there
2. Save .mxd to this same folder
3. If not already present, add ArcGIS Server 10.1 to Connected Servers in ArcGIS Desktop
4. Confirm data is stored in the correct location, share map as service on ArcGIS 10.1 server
5. Ask professor to “unlock” service for public viewing

Application creation:

1. Go to <https://www.arcgis.com> (or <https://#####.arcgis.com>)
2. Create account
 - a. If professor has created account for you, confirm account and change password
 - b. If professor has not created (and will not be creating) an account for you, create a public account. These have some limitations but are sufficient for creating basic, public web maps.
3. Follow instructions to create an ArcGIS.com hosted application
4. Add services to application using service URLs generated previously, replace <http://####.###.###.###:6080> with <http://gis.uncw.edu/>
5. Configure web map as desired, adding basemap, map tools, popups, etc.

Basic user:

6. Publish finished application to arcgis.com to public-facing network URL: http://#####.arcgis.com/apps/<generated_by_arcgis.com> where “<generated_by_arcgis.com>” is generated by the site

Advanced user:

6. Download provided template
7. Load template (ideally) in IDE or text editor (Aptana is used in instructions in Appendices)
8. Modify template to point to ArcGIS.com web map hash id

9. (For users familiar with JavaScript and Web Development only!) Modify JavaScript to customize presentation and add desired functionality
10. User (usually professor) with credentials copies modified application folder and files to //gis.uncw.edu for web server deployment
11. Application will be viewable at public-facing address: `http:// //gis.uncw.edu /arcgis/<application_name>/` where “<application_name>” is the user-provided name for the application

3.2.1.5 System Component Analysis

As previously highlighted, there are currently 3 different ArcGIS servers administered by UNCW's department of Geography & Geology: versions 9.2, 10.0 and 10.1. A summary of comparable components across these three servers is provided in the table below:

Similar Components Across Servers			
Server Version	9.2	10.0	10.1
GIS Server Components	Server Object Container & Server Object Manager	Server Object Container & Server Object Manager	GIS Server
Web Server	ADF Web Server for Services & Applications	ADF Web Server for Services & Applications	ArcGIS Service Web Server + Apache Tomcat Webserver for Applications or http://uncw.arcgis.com
Service and Application URL	Public facing web address: http://gisweb.uncw.edu	UNCW network-facing address: http:// ###.###.###.### :8399	Service: UNCW network-facing address: http:// ###.###.###.### , Application: https://#####.arcgis.com
Data Store	Network-based data store	Local data store on server	Local data store on server
Machine (Image) Model	ESX 4.1 virtual machine	ESX 4.1 virtual machine	ESX 4.1 virtual machine
Processor	Intel® Xeon® CPU E5640 @ 2.67 GHz	Intel® Xeon® CPU E5640 @ 2.67 GHz	Intel® Xeon® CPU E5640 @ 2.67 GHz (2 Processors)
Memory	2.66 GHz, 2.00 GB of RAM	2.66 GHz, 3.00 GB of RAM	2.66 GHz, 4.00 GB of RAM
Hard drive capacity	49.8 GB	39.9 GB	39.8 GB
Network card	VMware Accelerated AMD PCNet Adapter	Intel® PRO/1000 MT Network Connection	Intel® PRO/1000 MT Network Connection
Operating System	Microsoft Windows Server 2003, Standard Edition, Service Pack 2	Microsoft Windows Server 2003 R2, Enterprise X64 Edition, Service Pack 2	Microsoft Windows Server 2008 R2, Enterprise (64-Bit)

Figure 19: Comparison of comparable components between ArcGIS Servers 9.2, 10.0 and 10.1 as installed for the UNCW department of Geography and Geology.

The components in each of the three installed ArcGIS Server systems are discussed for relative comparison. Greater emphasis has been placed on ArcGIS Server 10.1 for analysis and alternate configurations since service and application development will be continuing on this server moving forward. The primary goal for the Analysis & Design is to develop 3-4 alternative system configurations for the ArcGIS 10.1 Server system. To determine the most viable configurations, a combination of literature review, staff and faculty interaction, and on-site research was used. For component (and sub-component) definitions, see section 2.3.1.

3.2.1.5.1 GIS Server

In ArcGIS Server versions 9.2 and 10.0, the GIS Server was sub-divided into the Server Object Manager (SOM) and Server Object Container (SOC) (See 2.3.1 for definitions). In both the 9.2 and the 10.0 installations, the SOM and the SOC were given a single-machine installation. If UNCW had required a more distributed workload, the installer could've installed multiple SOCs across different machines, each capable of running multiple services, all managed by one SOM.

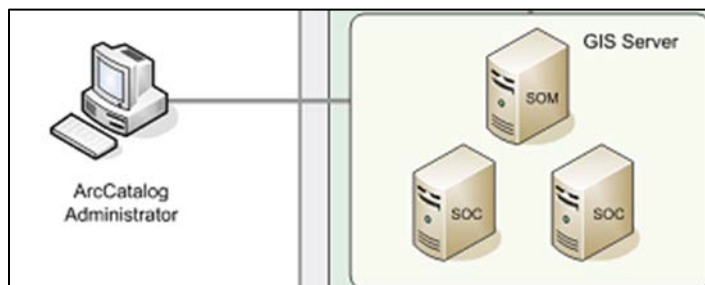


Figure 20: An example SOM/SOC configuration (Esri n.d.)

As of ArcGIS 10.1, Esri did away with the user-configured SOM/SOC model (they still run behind the scene but the user no longer configures them) electing instead to just simply have a GIS Server. For a similarly distributed workload, a user can now install copies of ArcGIS Server 10.1 across multiple machines which then work together to provide services. Additional installations beyond the first require a reference to the original machine which can be specified during the installation process (Esri 2013).

The analysis of the GIS Server component was drawn primarily from literature review since Esri and VMware have already teamed up and tested a very similar configuration (section 2.3.2). Increasing the number of GIS Server installations across multiple machines would distribute workload however the efficiency would be secondary to increasing the amount of RAM and number of CPUs available to one virtual machine. These tests indicate that increasing the number of GIS Servers should be considered only after increasing a single machine's number of CPUs and RAM. Further tests of the GIS Server's ability to handle its current workload were conducted using performance monitoring tests further along in sections 3.2.1.5.6 and 3.2.1.5.7.

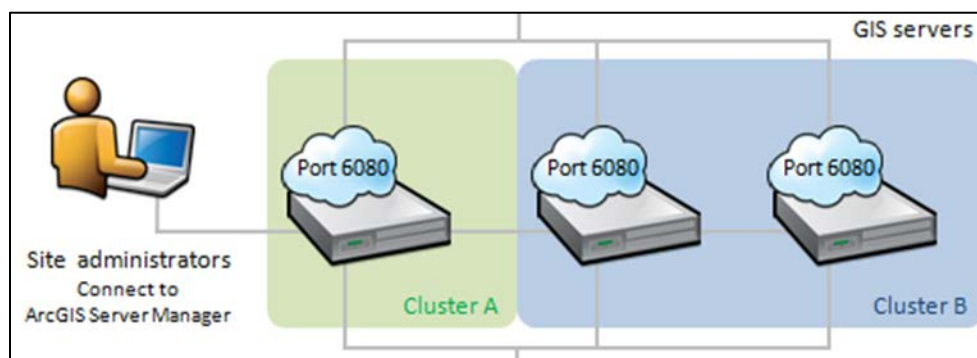


Figure 21: Example 10.1 installation entailing multiple GIS Servers (Esri 2012)

3.2.1.5.2 Web Server

ArcGIS Servers 9.2 and 10.0 used the Java Web Application Development Framework (ADF) which includes a built-in web server; this was installed as part of the ArcGIS Server installation package. The ADF Web Server is capable of hosting both services and applications. It can be installed on a separate machine, however in UNCW's case it was installed on the same machine as the GIS Server for both the 9.2 and 10.0 version servers.

In contrast, ArcGIS Server 10.1 provides a built-in web server for interacting strictly with services, not applications. As of version 10.1, applications are no longer hosted by a utility provided with the GIS Server package and therefore must be deployed using a separate web server. To most closely approximate the configuration of the 9.2 and 10.0 installations, an instance of Apache Tomcat 7.0.42 was temporarily set up on the same

VMware instance as ArcGIS Server 10.1 for hosting of applications. This Web Server was chosen as it met the following criteria:

- It was freely available for use under the terms of the Apache License, Version 2.0
- It's Java compatible (Dr. Halls' preference) and, as of the date of installation (September 27, 2013), supported Java version 1.6 or later. Installed Java version was 1.7.0_15.
- It was supported by the ArcGIS Web Adaptor (See section 2.3.1 for definition, also explained further below). The ArcGIS Web Adaptor supports version 7.0, 7.0.42 was the latest version available at the time of installation (September 27, 2013).

The Web Adaptor required separate installation but was easily configured to use Apache Tomcat 7.0. It should be noted that Dr. Halls' preference for a Java-compatible web server likely stemmed from the requirements of Java Web ADF of previous version. The applications currently being built by the students and staff don't require Java compatibility therefore IIS would be a sufficient as well. There are a number of different web servers listed as compatible with the ArcGIS Web Adaptor. The biggest factor in the case of the web server is not performance or compatibility but rather maintainability by ITSD moving forward. Discussions with UNCW's ITSD revealed they were capable of working with IIS version 7 and Apache-Tomcat 7.x.

Given their preference, ITSD chose to install IIS 7 on application web server. To install the ArcGIS Web Adaptor, it was also necessary to install the .Net 3.5 Framework. It was also noted that unlike Apache-Tomcat 7, IIS was not immediately capable of hosting mp4 content, required for one of the migrated applications. Configuration using the IIS Management interface allowed for this simple adjustment.

3.2.1.5.3 Service and Application URL

Services and applications published on the ArcGIS 9.2 Server are accessible via the host name: <http://gisweb.uncw.edu> as outlined in section 3.2.1.2.3. The network address for the server deploying map applications is “//#####.uncw.edu”. The difference between the server's address and the URL indicates an alias set up on the UNCW web server for a

publicly-facing URL. This is an ideal setup as UNCW's internal host name and IP address are obfuscated.

Services and applications published on the ArcGIS 10.0 Server were accessible via the host name: `http:// ###.###.###.###:8399` as outlined in section 3.2.1.3.3. Unlike the version 9.2 server, this server was not configured for public access and was only accessible within the UNCW network. Applications deployed on this server could only be accessed using the host name and port number.

Services deployed on the server hosting ArcGIS Server 10.1 were initially accessible directly via the host name: `http:// ###.###.###.###:6080` as outlined in section 3.2.1.4.4. During the trial run with Apache-Tomcat 7, applications were accessible using the host name: `http://server02.uncw.edu`, though only within the UNCW network. This configuration was later changed as a new VM Web Server was setup and the web adaptor installed on it. This had the following effects:

- Services could be accessed through the web adaptor using the URL:
`http://gis.uncw.edu/arcgis/rest/services/<folder_name>/<service_name>/`
- Applications hosted on the web server could be accessed using the URL:
`http://gis.uncw.edu/<application_name>/`

It was necessary to work with ITSD to secure an exception for this server to make it publicly accessible. While the 10.0 server didn't require a web adaptor, this configuration was required for public access of services and applications on the 10.1 server. This configuration most nearly mirrored the version 10.0 server's configuration and actually improved upon it since it doesn't require a port # in the URL.

3.2.1.5.4 Data Stores

Application data for the ArcGIS 9.2 Server was stored on a shared network drive accessible to both the virtual machine hosting the ArcGIS server, and faculty and students with the correct permissions. In order for ArcGIS Server 9.2 to access folders via UNC path names, the service running ArcGIS Server had to be given privileges to this folder. To accomplish this, the domain user name `#####@uncw.edu` was provisioned by ITSD for

use by the ArcGIS Server SOC process. This user name was given read\write\modify privileges to the NAS-based webgis folder and all descending folders.

Similar to the ArcGIS 9.2 Server, application data for the ArcGIS 10.0 Server was stored on a shared network drive accessible to ArcGIS Server service, and faculty and students with the correct permissions. In order for ArcGIS Server 10.0 to access folders via UNC path names, it had to be given privileges to that folder. To accomplish this, the domain user name #####@uncw.edu, originally provisioned by ITSD for use by the ArcGIS 9.2 Server, was used to run the SOC process. This user name had already been given the correct read\write\modify privileges to the NAS-based webgis folder and all descending folders when the version 9.2 Server was originally set up.

The ArcGIS 10.1 Server was initially set up with only a local data store. This was mostly due to timing as our primary contact in ITSD left his position at UNCW shortly following the initial server setup. From April, 2012 through September 25, 2013, service and application data was stored locally on the virtual machine, accessed using a local account provisioned for the ArcGIS Server service. On September 25, 2013 the service was reassigned to run under the #####@uncw.edu domain account used by both the 9.2 and 10.0 version servers. Though the name bears little relevance for the current server (there's no longer the notion of a SOC as far as the user is concerned), the configuration and privileges of the account were still ideal for the current server account. As with the previous two servers, use of this user name allowed the ArcGIS Server service read\write\modify privileges to the NAS-based webgis folder and all descending folders.

In the past, ITSD had advised Dr. Halls that it would be easier to keep map data in a network location as it was easier to add storage space to the network drive. More recent interactions with ITSD indicated adding extra space to either location would be a viable option. This shifted the analysis from a focus on maintainability to performance instead. CrystalDiskInfo 5.6.2, a free utility available from Hiyohiyo, was used to test read/write speeds of both drives at random times over the course of several days. The following test settings were used:

- # test runs: 6

- # runs per session: 5
- File test size: 500 MB
- Drives tested:
 - Local: C:\
 - Network: \\arcshare\GIServer\public

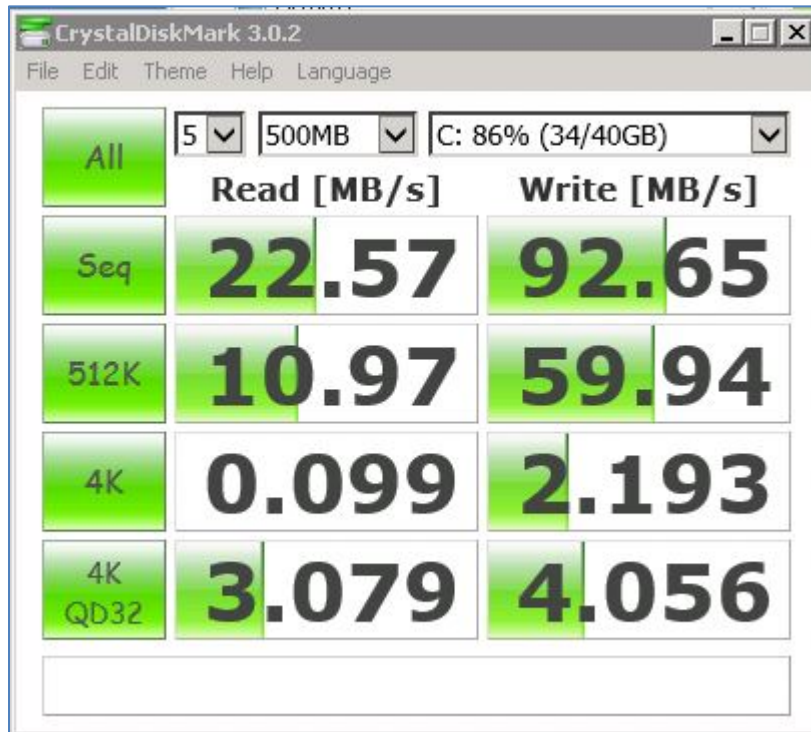


Figure 22: CrystalDiskMark, the utility used for read/write speed assessments. Sequential read/write tests were conducted followed by random read/write test comprised of blocks in sizes of 512KB, 4KB and 4KB with Queue Depth of 32.

The results obtained from these tests are discussed in Section 4.

3.2.1.5.5 Machine (Image) Model

Each of the three virtual machines ran on VMware's vSphere 5.1 model. As there is no substantial difference between each of these virtual machines, there was little need for discussion on differences between each. Inquiries with UNCW's ITSD indicated that VMware models are upgraded periodically as new licenses are acquired. It's useful to note

that the current 5.1 model is the same VMware model used in Esri's and VMware's joint testing conducted by Bhattacharjee et al. (2013). Enhanced performance for this virtual machine model over previous models was noted and discussed in the literature review in Section 2.3.2.

3.2.1.5.6 CPU

Each of the VMware machines hosting ArcGIS Servers used the Intel® Xeon® CPU E5640 @ 2.67 GHz; the ArcGIS 10.1 Server actually uses 2 of these CPUs as configured by ITSD. The launch date for the processor was listed as "Q1, 2010" (Intel, n.d.). This launch date indicated the ArcGIS 9.3 Server machine had been upgraded following installation since this occurred in 2007. According to Intel (n.d.), the CPU is a quad-core, each core capable of handling up to 8 threads.

The model of the CPU on the server hosting ArcGIS 10.1 had proven sufficiently capable of running services so far. There was no official Esri-required CPU architecture at the point of research ("System Requirements", 2013) since there are a number of different factors that could affect the speed of the system. Nevertheless it was an encouraging sign that the model of the processor was identical to that used in the benchmarking tests performed by Esri and VMware (Sakowicz & Pizzi, 2012). Their tests of the VM environment indicate that the CPU was very capable of handling services and showed strong performance increases as the number of CPUs available to ArcGIS Server on a single machine was increased. The conclusion may be drawn from these results that the focus of performance enhancements should be more concerned with the number of sufficiently-capable CPUs made available to the server rather than a comparison of different CPUs. Esri did provide a link to <http://www.SPEC.org> for CPU speed comparisons if specific CPU speed is a focus of future analysis.

While CPU speed wasn't a primary focus, periodic use of the system's standard performance monitoring software on the ArcGIS 10.1 Server was nevertheless performed. Usage spikes between 10-75% of CPU capacity were observable during query requests for the rebuilt Riverrun and Oculina applications. Since the server currently operates on two CPUs, these spikes only affected one of the two CPUs.

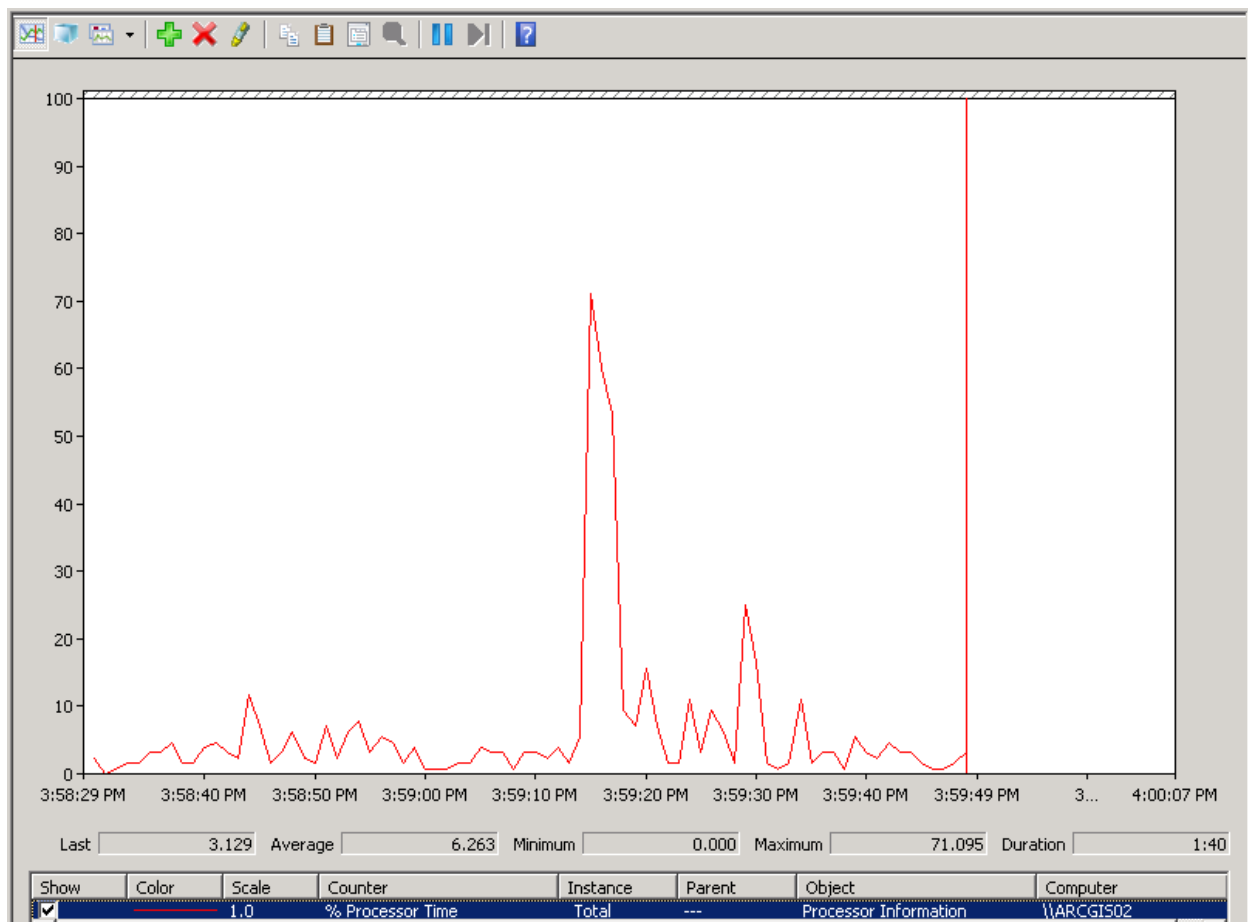


Figure 23: Performance during requests for Riverrun and Oculina data.

3.2.1.5.7 Memory

Each of the three servers used 2.66 GHz RAM; the web GIS servers hosting ArcGIS Server 9.2, 10.0 and 10.1 had each been allocated 2.00 GB, 3.00 GB, and 4.00 GB respectively. Esri suggests a minimum of 4 GB of RAM for an ArcGIS 10.1 Server environment given the following typical setup (“System Requirements”, 2013):

1. Two cached map services
2. One dynamic map service
3. One locator service
4. One geoprocessing service
5. Geometry service
6. PrintingTools service

7. SQL Server Express
8. IIS with Web Adaptor

While the 4 GB memory in //server02 was an improvement over the previous servers, it likely won't be sufficient to handle the expected volume of traffic; one GIS class alone might produce up to 30 dynamic services. There are ways to mitigate the memory footprint such as caching map services (#1 in the list above); these are discussed further in Section 4. It seemed likely the server would require a greater amount of memory.

Memory consumption was minimal during query requests but much greater during highly graphical operations serving raster data. During this, overall memory consumption was observed to occasionally use the entire allotted 4GB. This behavior was less likely during serving of vector data.

3.2.1.5.8 Hard drive capacity

The web GIS servers hosting ArcGIS Server 9.2, 10.0 and 10.1 were noted as having 49.8 GB, 39.9 GB and 39.8 GB capacities respectively. ArcGIS Server 9.2 has access to approximately 10 GB greater local disk space vs. the other two servers however it was also noted that the majority of this goes unused. The web GIS Server (//server01) hosting ArcGIS 10.0 hosted the greatest number of applications (~21) at the time of writing (October 8, 2013), however its disk usage was less than 20 GB. The relative low disk usage of //server01 was likely attributable to the use of a network data store for primary storage of service and application map data.

ITSD indicated that providing greater storage to either the virtual machine hosting ArcGIS Server 10.1 or the network drive shouldn't present any difficulty. During the analysis of the drives, //arcshare was actually increases from 2 TB to 3TB following an incident where space was exhausted. This made storage capacity in either location less of an issue, and speed of I/O (Section 3.2.1.5.4) and administration the greater considerations. Speaking with a representative from ITSD, I was able to confirm that space on drives is increased as needed (usually by request).

3.2.1.5.9 Network card

The web GIS Server hosting the ArcGIS 9.3 Server had been allocated the VMware Accelerated AMD PCNet Adapter. The web GIS Servers hosting ArcGIS Servers 10.0 and 10.1 had been allocated the Intel® PRO/1000 MT Network Connection. The two adapters were the VMware defaults selected based upon the operating system architecture; the VMware Accelerated AMD PCNet Adapter is allocated to 32-bit systems whereas the Intel® PRO/1000 MT Network Connection is allocated to 64-bit systems. ITSD confirmed the VMs were connected to a 1GB Management Network & 10 GB VLAN to clients.

3.2.1.5.10 Operating System

The three web GIS servers used the following operating systems:

- //uncwwebgis (ArcGIS Server 9.2): Microsoft Windows Server 2003, Standard Edition, Service Pack 2
- //server01 (ArcGIS Server 10.0): Microsoft Windows Server 2003 R2, Enterprise X64 Edition, Service Pack 2
- //server02 (ArcGIS Server 10.1): Microsoft Windows Server 2008 R2, Enterprise (64-Bit)

The servers hosting ArcGIS Server versions 10.0 and 10.1 both benefited from 64-bit OS architectures capable of working with greater than 4 GB RAM which in the long run should provide a more flexible solution. ArcGIS 10.1 requires a 64-Bit architecture so the choice of Microsoft Windows Server 2008 R2, Enterprise (64-Bit) is an appropriate one.

ArcGIS Server supports several operating systems however the following quote was found in “System Requirements” (2013), “Typically, a software component will run faster in the environment [in which] it was natively built.” Since services and applications will be built in a Windows Environment, Windows Server 2008 should be a strong option. ITSD confirmed it will support either Windows or Linux if necessary.

3.2.2 Service & Application Migration

There were forty ArcGIS Server applications that were only compatible with the old server environment; of these forty applications, two were designated as critical for migration to the new server.

- Riverrun: An Interactive mapping application for visualizing water quality parameters in the Lower Cape Fear River.
- Oculina: An Interactive mapping application examining the ‘Oculina Habitat Area of Particular Concern’, a deep-water coral reef area off the Atlantic coast of Florida.

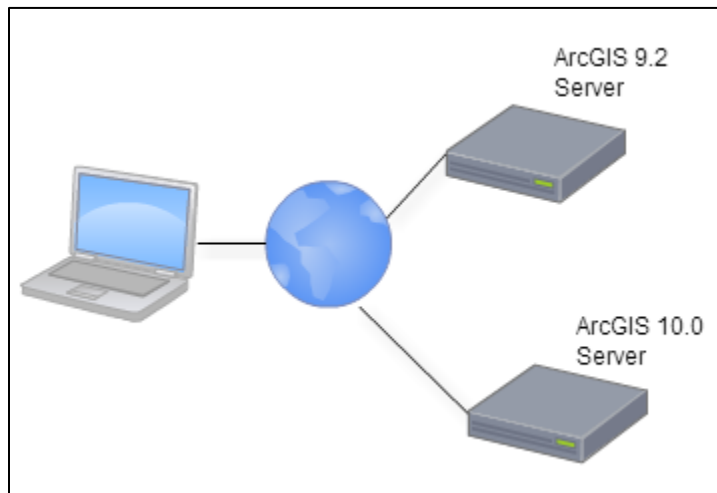


Figure 24: The GIS portal originally pulled data from 2 ArcGIS Servers, versions 9.2 and 10.0

Specific elements of each map service were to be preserved through migration, a table listing elements that had been identified for potential migration can be found in Appendix D. The remaining applications on the old server were not deemed critical for migration per Dr. Halls and therefore were not considered deliverables for this project.

GIS Applications on the old server were written using ArcIMS Web ADF, an Esri application development framework for Java Platform Version 9.2, SP 3. This information was derived from the projects’ metadata files. Details regarding the method by which the projects were built were significant as they impacted the code changes required for successful migration. Though migration was later proved to be impractical (we’d focus more on rebuilding the applications), the following initial steps were attempted for application migration:

1. Worked with Dr. Halls to determine what course(s) of action were taken in previous attempts to migrate the applications. Whenever possible, tried to determine the previous failure modes encountered.
2. Confirmed the list of critical applications to be migrated with Dr. Halls.
3. Obtained remote access and read/write privileges to UNCW's ArcGIS server from ITSD.
4. Created a copy of all projects for code testing and modification.
5. Confirmed projects had been developed using the ArcGIS API for Javascript. The difference between APIs used to develop the projects wouldn't substantially impact the time required to make any necessary changes however it would affect the code changes that needed to be made. Therefore it was necessary for to confirm the ArcGIS development API version used to create each project using either assistance from Dr. Halls or each project's metadata in order to implement the necessary code changes and restore each application's functionality.
6. Loaded each project locally in a web browser to check compatibility of applied Javascript changes. Used posted projects on old server as a baseline for comparison. If project failed to load correctly, determined failure mode and modified code as necessary.
7. Created a test version of the GIS portal. This would serve as an unlinked copy of the current site to be used for testing of posted projects.
8. Posted projects to test portal and confirmed Javascript functionality as well as Java web-server compatibility. Used posted projects on old server as a baseline for comparison. If project failed to load correctly, determined failure mode and modified code as necessary.

Since a new server installation was required for this project, no applications had been deployed prior to this migration. The steps taken for ArcGIS Server's installation were sufficient to allow for deployment of services. Services provide the supportings data for applications. The process of preparing the server to accommodate application development is detailed in the next sections.

3.2.3.1 Migration Tools

For the Service & Application Migration, the following tools were used:

- Microsoft© Windows™ 7 64-Bit OS
- Google© Chrome™ Version 29.0.x
- Microsoft© Internet Explorer versions 9 and 10
- Mozilla Firefox version 17.0.x
- Cisco© AnyConnect™ VPN Client will be used for remote access to the UNCW network
- ESRI© ArcGIS Server™ Enterprise Edition 10.1
- ESRI© ArcGIS Server™ Enterprise Edition 9.2
- ESRI© ArcGIS Desktop™ 10.1
- Notepad++ Version 6.3.x, an open source editor for basic code view and modification
- Aptana Version 3.0
- Oracle© Java™ Platform SE 7u7
- Cygwin 1.7.22 (for git version control)

3.2.3.2 Riverrun Service Migration

Riverrun was a service built upon the ArcIMS platform which was found as of ArcGIS Server 10.1, to no longer be supported. For this reason, the request to port the project became less a migration request and more of a project-conversion request. ArcIMS was a product previously offered by Esri to serve maps over the internet. This technology was abandoned in favor ArcGIS Server services which provided more customizable maps with interactive capabilities that leveraged ArcGIS tools and extensions (Carnes 2004).

ArcIMS map layout files were stored in a proprietary .xml-style format with the extension “.axl”. There appeared to be no officially supported method for migrating from ArcIMS to a modern application format as of March, 2013. Also missing was the original ArcGIS Desktop (.mxd) source file. A conversation with Dr. Halls indicated there may never have been one since early services could be created directly in ArcGIS Server 9.3 with no ArcGIS Desktop file. There were however a few potential 3rd party solutions for creation of .mxd source files from ArcIMS data. My attempts with these solutions are outlined below:

Deriving source-data attempt with ‘AXL2MXD’:

1. As in section 5.1.1, all files could be found one central directory. Prior to any operations on these files they were copied to a new directory which is referred to here as “%riverrun_root%”
2. My first attempt used the “AXL2MXD” script, a user-published script posted by Nianwei Liu at arcscrippts.esri.com. This script came in both a standalone windows version and a Dos command line version.
 - a. Windows version: Resulted in unhandled exception with the option to ignore the error and continue. Both ignoring the error and continuing failed to result in file conversion
 - b. Dos version: Running the executable prompts the user to supply the name of the .axl file as the first argument. After providing this argument, running the executable with the name of the .axl file triggered a “file not found error” as it attempted to locate a required resource ‘ESRI.ArcGIS.Carto, Version 9.2.0.1324’.

Deriving source-data attempt with ‘GDK ArcTools’:

1. The second attempt employed the script “GDK ArcTools”, found at arcscrippts.esri.com and originally posted by Carsten Helsted. In the script’s descriptive summary, it claimed it was compatible with ArcGIS 9.2 and 9.3. The version of ArcGIS being used for this project was 10.1 nevertheless installation was attempted. The executable gave no indication of failure however the script could not be found as a toolbox, extension or add-on following installation. Since there was no associated help content and the most recent version of the software was over 4 years old, it was difficult to determine why the execution had failed. It seemed likely that it required ArcGIS to first open ArcIMS files. Since ArcGIS 10.1 is incapable of working with ArcIMS files, there may have been a compatibility issue.

A question asked at the 2009 Esri International User Conference also covered the issue of migrating from ArcIMS to ArcGIS Server; the following statement was made in response: “...we encourage ArcIMS users to adopt new workflows for creating effective Web

maps.” (Esri 2009) Given this response, it seemed likely that the best option would be to recreate the IMS service from scratch using the same supporting data.

An email exchange with Dr. Halls confirmed the approach of starting building the maps from scratch as the best approach; the remaining steps detail this process.

The following table summarizes the folders and files originally found in %riverrun_root%:

Folder	File types contained	Notes
Covs	.adf files	
IMS	.axl files	
Legends	.avl files	An <i>ArcView GIS 3</i> legend file. Will retain but may or may not be necessary.
Projects	.apr files	An <i>ArcView GIS 3</i> project file. Retained for use in data import.
Riverrun	.class, .js, .htm	IMS files specific to IMS publishing. Not retained since web-publishing process will be different.
Sampledata	.txt	Tab separated text data files containing map supporting data.
Shapes	.dbf, .shp, .shx	Shapefiles, retained.
Worddocs	.doc	Two MS Word documents describing the project

Figure 25: File-types found in %riverrun_root%

Deriving source-data attempt with ‘Import from ArcView Project’:

ArcGIS had an available tool “Import from ArcView Project” for importing .apr files to the current 10.1 desktop. I attempted this with the largest and most recently modified .apr. The tool was able to successfully import the .apr data symbology and structure however none of the supporting geographical data was present. This was likely for one of two reasons: file paths in the .apr were absolute and the folder’s been moved since it was

created or the folder structure has been changed in some way with folders possibly renamed or moved around. An examination of the other included folders showed that many of the supporting data was still present and could be re-mapped which I confirmed with the first layer “fishmerged”.

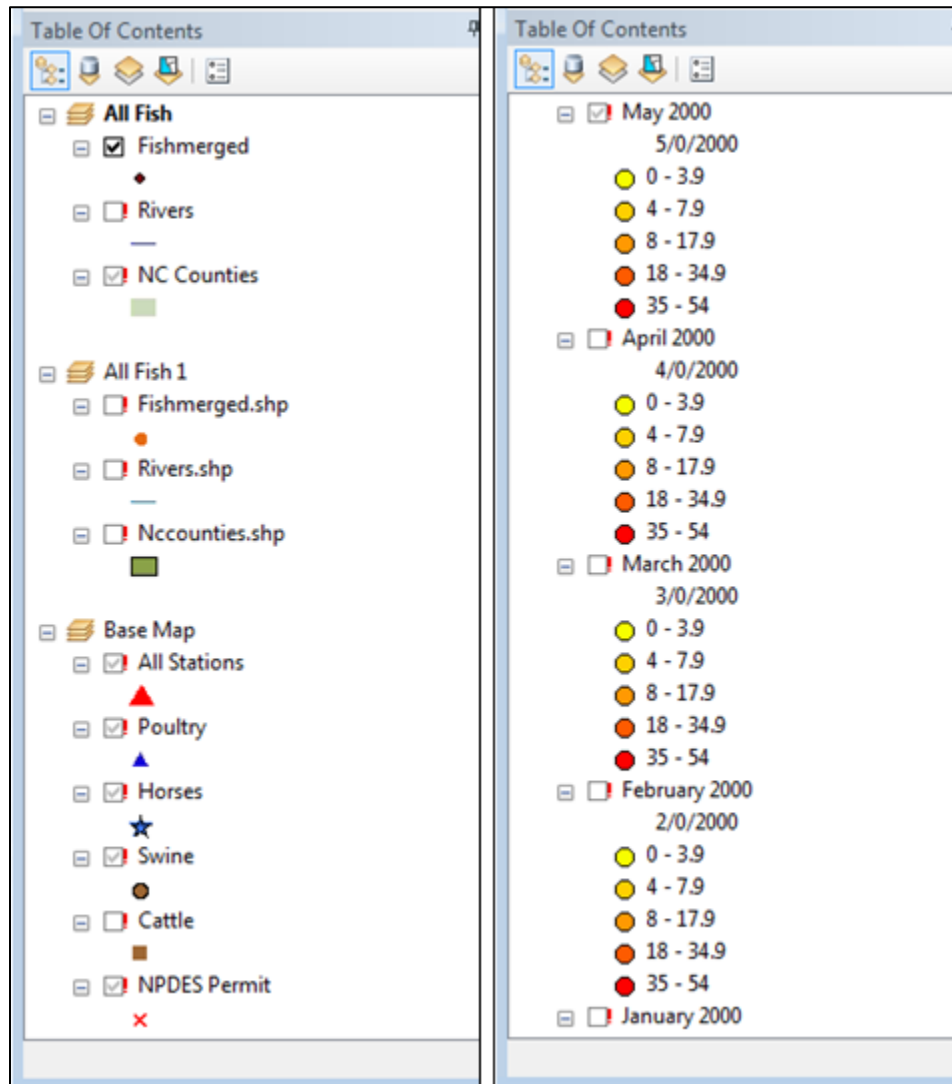


Figure 26: Data layers missing source data, Fishmerged remapped to source

The most practical course of action proved to be remapping the paths to each of the “fixed” layers manually (changing the path to reflect the new file location). Here the term “fixed” refers to the fact that these layers don’t change as a function of time. Several other

“non-fixed” layers however do change as a function of time. The following illustration better describes the allocation of data:

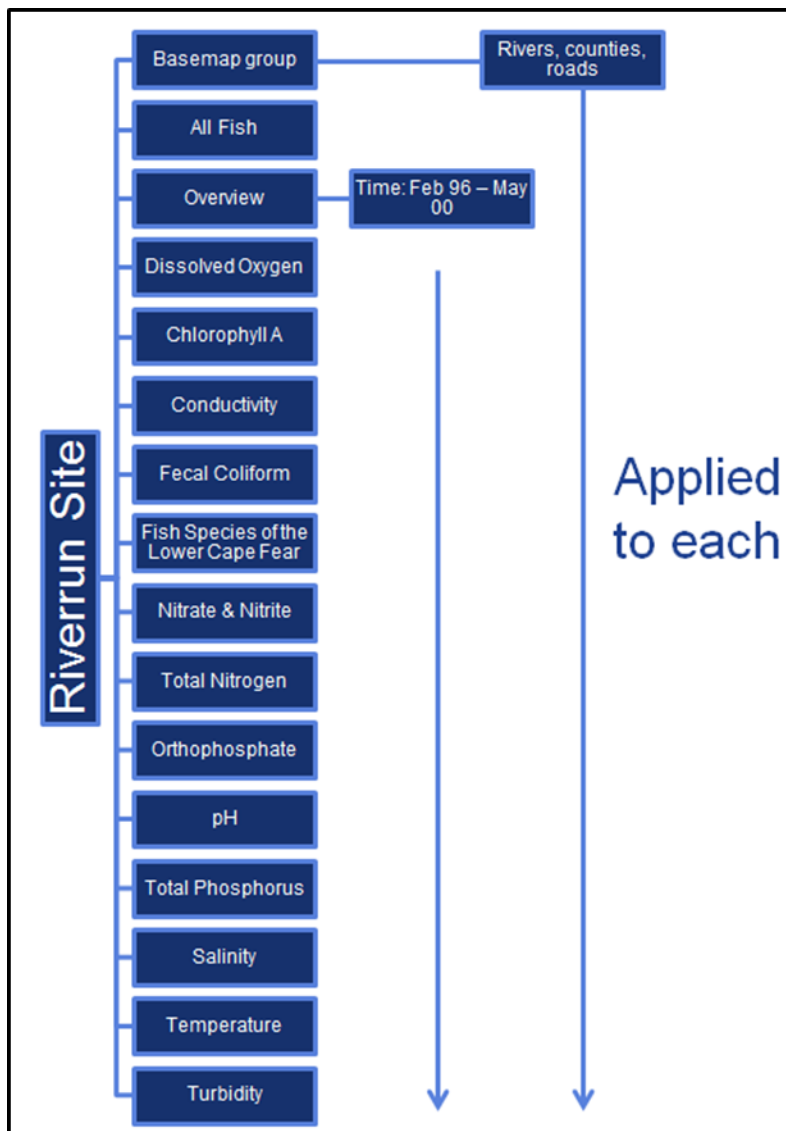


Figure 27: All but two layers changed as a function of time, all layers required a basemap.

The .apr import was useful in disclosing details regarding the symbologies and value ranges used however it wasn't clear how data was originally extracted from each shapefile. Nevertheless the goal of each non-fixed shapefile was clear: to display data relative to the collection date.

The first attempt to publish a map service was hampered by the supporting data. Combining the layers in the data frame revealed two facts:

- Some projections weren't defined
- Coordinate systems were different for different layers

These problems were easily resolved as the layers were projected correctly, they just needed to be defined using the 'Define Projection' utility in ArcGIS desktop.

3.2.3.3 Oculina Service Migration

At the time of the migration, I discovered I was unable to access the 9.2 server. I was however able to access the files used to create the service as well as a non-working shell of the original service accessible via the GIS portal. Given the lack of Server 9.2 access and the age of the project (most files had not been altered since 2007), the first attempt at migration was an attempt to recreate the service using the original source .mxd file (which in the long run, proved to be the better option anyway). Unlike Riverrun, Oculina had been built using ArcGIS Desktop and therefore had an .mxd file which I could use for publishing a service.

The Oculina project's GIS source files were located in the "%public GIS folder%\webgis\oculina" directory created by Dr. Halls. The steps taken to publish the service on the 10.1 server from here forward used a copy of the files, located at "%public GIS folder%\webgis\oculina_10_1" which from here forward is referred to as "%oculina_root%".

Service republishing attempt using original .mxd file:

1. From ArcGIS Desktop, the file "%oculina_root%\oculina_gis\ocuina_gis_web.mxd" was opened.
2. This file was then republished to the new server using the instructions provided by Esri (Esri 2013)
3. Upon completion of the publishing process, the service was confirmed to run as a REST javascript service at the following location:

http://###.###.###.###:6080/arcgis/rest/services/admin_projects/oculina_gis_web/MapServer?f=jsapi.

3.2.3.4 Oculina Application Migration

It was confirmed in Esri's literature (Esri, 2011) that ArcIMS applications could not be deployed on ArcGIS Server 10.1. This project found itself on the border of a substantial architecture change for ArcGIS. When the project idea was established, the applications were instead going to be migrated to ArcGIS Server 10.0 which was capable of hosting ArcIMS applications. As of 10.1 this was no longer the case. Because of this fact, it was clear that moving the applications to the new server would require an application rebuild using currently supported technology.

Since Oculina was the first application to be rebuilt, there were several questions that needed to be answered. The answers to many of these questions could be applied to other applications developed on the 10.1 server as well. The first of these questions was: "Which API would we be developing in?" As of version 10.1, Esri offered three primary APIs for web application development. The following is a table of the APIs that were supported by Esri along with their respective benefits for UNCW's Department of Geography & Geology:

API	Benefit	Appropriate for this installation?
JavaScript	Flexible, requires only browser on client side.	Yes, JavaScript is a flexible cross platform solution.
Flex	Provides a utility for creating maps quickly in a code-free environment.	Maybe. The code-free aspect would be appealing for non-technical people however final map applications require client to have Adobe Flash Player 10. It was however discovered that Adobe is no longer developing Flex technology.
Silverlight	Leverage Microsoft's utilities Visual Studio 2010 and Expression Blend 4 for visually impressive maps.	No. Students will not have access to Microsoft coding utilities (such as Visual Studio). This approach would be outside of the current scope for GIS instruction.

Figure 28: Esri's available APIs for application development and respective advantages/disadvantages.

After conversing with Dr. Halls and discussing several critical factors outlined in Figure 28, it was determined we would move forward with JavaScript development.

I chose Aptana Studio 3 both due to my familiarity with the IDE and for the following benefits it provided:

- It had a common development layout similar to Eclipse or Visual Studio
- It had a built-in web server for testing
- It was git compatible.

I chose git for source control, designating my Google Drive as my remote repository for pushing updates. This setup was reasonable for a scenario in which one person is developing on a project but would likely be less reasonable for team development. My local Google Drive folder would auto-syncs within seconds of receiving updates thus backing up my code immediately. This was a relatively safe operation however I acknowledged it's less safe than having a more traditional remote git repository.

Within the ArcGIS JavaScript API it was at first unclear whether it would make more sense to develop an application using one of Esri's many provided JavaScript templates or to instead start from scratch. I ended up trying both but quickly abandoned the template-approach for two reasons:

1. Templates were primarily geared towards applications that consumed web maps from ArcGIS Online
2. They allowed the developer to take several things for granted in terms of UI elements and the dojo framework which were aspects of development I wanted more experience with

To expound further on #1, there were two different ways to "get" maps in an ArcGIS JavaScript application. You could either:

1. Use the API to build a map and add layers directly from services
2. Use a hash id from a web map built in ArcGIS Online. Much of the details of the map and layers are obscured in this process though it still leverages the ArcGIS JavaScript API

Using the API to build and add layers directly from the services you've created (or other publicly available services) allows greater flexibility in how you present, query and control your data. It also eliminates the need for an ArcGIS Online account. This is less relevant in our situation since ArcGIS Online came with our site license, however in the broader sense it was good to understand both options. On the flipside, using the API and bypassing ArcGIS online required more work and it meant I would have to develop greater understanding of the nuances of using the JavaScript API.

The advantages of using ArcGIS Online web maps to load map data into the map application were primarily the quickness and ease of development. ArcGIS Online templates were tailored for this purpose and allowed a user to get a map up and running quickly. This was a very suitable solution for student application development however it didn't offer me the flexibility and options I needed to incorporate the highly customized functionality required of the Oculina application. Nevertheless its suitability for student development is revisited later in Section 3.2.4 for Sample Application development.

The following is an outline for how the Oculina application was developed:

1. Oculina was developed from scratch beginning with the UI. Esri recommended the Dojo Framework and had co-developed several widgets with Dojo. The basic structure of the application was laid out first.
2. UI elements were populated with non-negotiable elements which define most map applications such as a map and legend.
3. Dr. Halls requested a slider to change layer focus and change the current query layer. A Dojo slider was incorporated that altered the visibility of the layers and changed which layer gets queried when the user clicked the screen.
4. The previous iteration of the Oculina application had the ability to turn layers on and off, even when they weren't the focus of queries. I was able to provide similar capabilities in the form the freely available 'Table of Contents(TOC) Widget' developed by Nianwei Liu (Figure 29). This widget added some great functionality to the application however there were two things to note about its operation:
 - a. It produced two of all sublayers when an object of the ArcGISDynamicMapLayer class was added. This problem was avoided by feeding the application a layer array with a trailing null element.
 - b. The ArcGISDynamicMapLayer which contains the sub-layers that comprise Oculina's basemaps aren't visible in the legend in IE 9 or 10. They were visible in Chrome and Firefox. I discussed this with Dr. Halls, we concluded it was an acceptable compromise for the functionality it added to the app.
5. Adding layers using the FeatureClass class and the ArcGISDynamicMapServiceLayer class were tested across several layers. It was determined that layers on which queries would be performed, were better off added as objects of the FeatureLayer class while layers that required less user interaction were more easily added using the ArcGISDynamicMapServiceLayer class.
6. Query capabilities for each layer were tailored to show the most relevant information for that layer whereas before they were designed to show all fields. This was done in accordance with Esri query standards.
7. It was discovered that one query layer was intended to point users to a video for each dive site. Unfortunately that video had been encoded in MPEG-1, an outdated

video technology that's not supported by HTML 5. Videos were converted instead to the MPEG-4 format using the freely available application "Any Video Converter".

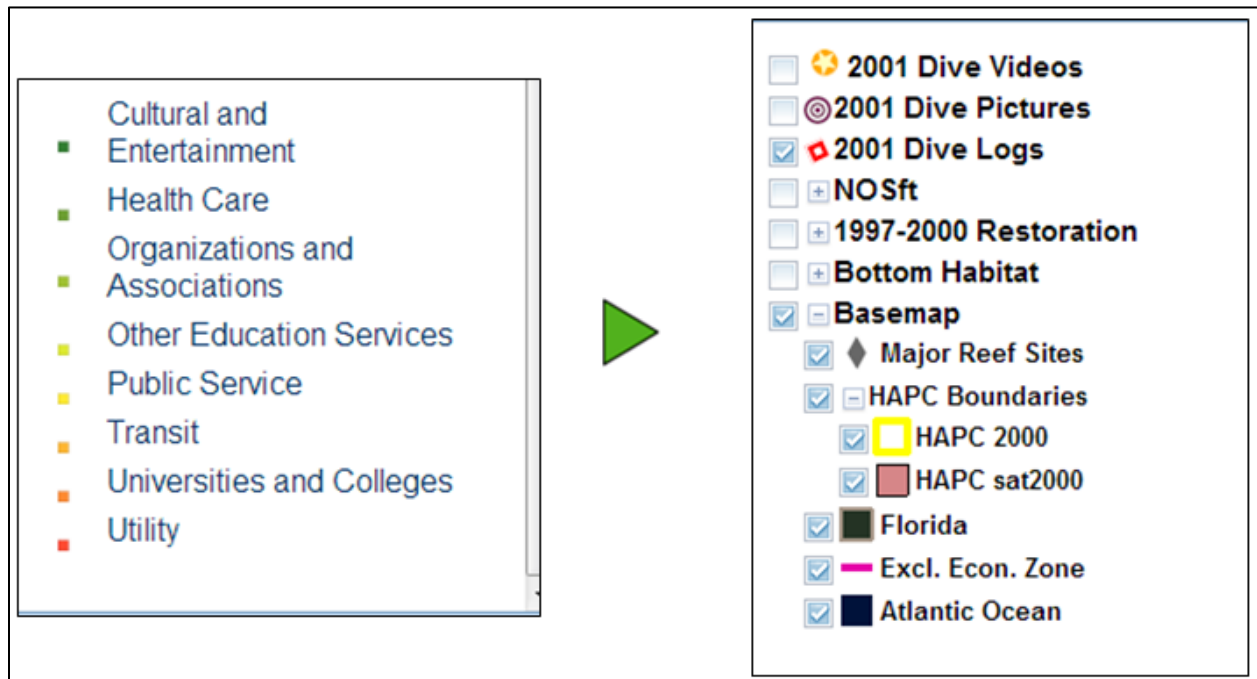


Figure 29: Legend improvement. An example of the normal style legend on the left vs. the Table of Contents(TOC) Widget on the right.

8. Several utilites from the old application were built using ArcGIS samples for reference including the following:
 - a. Distance measurement tool
 - b. Point buffer tool
 - c. Query layer by shape (rectangle)
 - d. Query by point
9. The Oculina application was then placed on //server03, the GIS application web server.
10. The final results of the Oculina application rebuild are covered later in the Results section.

3.2.3.5 Riverrun Application Migration

The rebuild of Riverrun built upon many of the lessons learned in Oculina. Though many aspect of this project were a copy/paste from Riverrun, the following list includes several aspect of this project which differentiated it from Oculina:

1. Whereas Oculina had been one site, Riverrun was effectively 14 sites which needed to be combined into one. This was done using multiple tabs.
2. Riverrun required working with larger datasets making it more critical to only show whichever layers were necessary
3. 12 of the 14 tabbed maps required a variable query that changed according to the selected date. This also needed to update the map and the legend.
4. Switching between tabs required completely removing the map and the legend and building them again with the new data for the new tab. Otherwise the map and legend had problems updating between tab switches. New routines had to be implemented to preserve the state of the map on each tab upon leaving so that it would be the same upon returning.
5. It was discovered during the Riverrun rebuild (though it applies to Oculina and other services and apps too) that the server had a bug which failed to respond to every other RESTful GET request (Esri bug designation: [NIM086349 – Alternate Query (Get) requests fail in Version 10.1 SP1]). The bug only occurred if the user was querying data stored in a File GeoDatabase on a NAS. This was consistent with our situation. This problem could be resolved by upgrading to the next version of ArcGIS Server (10.2) or by moving data to ArcSDE.
6. The Riverrun application was then placed on //server03, the GIS application web server.
7. The final results of the Riverrun application rebuild are covered later in the Results section.

3.2.4 Sample Application

3.2.4.1 Sample Application Tools

For the Sample Application Tools, the following tools were used:

- Microsoft© Windows™ 7 64-Bit OS

- Google© Chrome™ Version 29.0.x
- Microsoft© Internet Explorer versions 9 and 10
- Mozilla Firefox version 17.0.x
- Cisco© AnyConnect™ VPN Client
- ESRI© ArcGIS Server™ Enterprise Edition 10.1
- ESRI© ArcGIS Desktop™ 10.1
- Notepad++ Version 6.3.x
- Aptana Version 3.0
- Oracle© Java™ Platform SE 7u7
- Cygwin 1.7.22 (for git version control)

3.2.4.2 Sample Application Methodology

Dr. Halls indicated she'd like for the students to get experience developing Story Maps, an Esri initiative focused on telling stories using one of their provided map templates as a base for their project. I chose to build the sample application template after the two application rebuilds in order to incorporate lessons learned from each. As discussed in section 3.2.3.3, the best approach for a novice web developer would be to use the ArcGIS Online web map hosted approach.

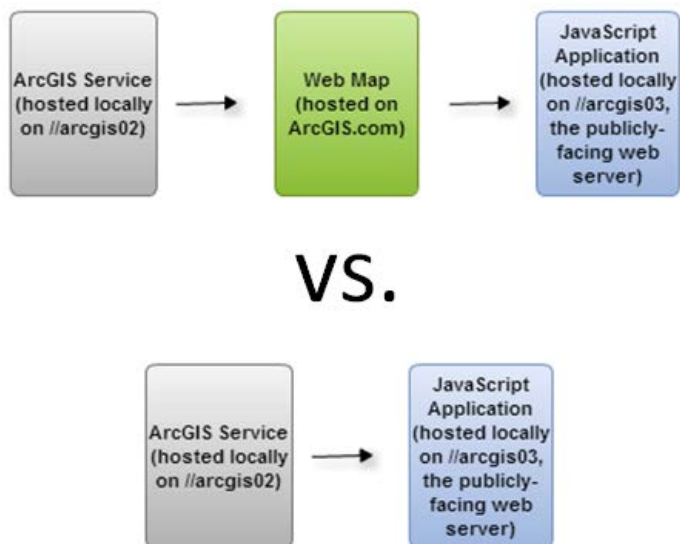


Figure 30: Adding the ArcGIS Online-hosted web map added an extra resource to keep track of and eliminated some customization but resulted in a far quicker application creation experience.

Dr. Halls and I settled on the “Storytelling Side Accordion – BETA” template, a customizable application made for hosting multiple web maps. For sample source data, Dr. Halls had a wealth of Masonboro map data available to populate multiple maps. Following the approach at the top of Figure 31, the following steps were taken to create the sample map application, details of these processes (with images) may be found in Appendices P thorough S:

Create Services:

1. A combination of Masonboro .mxd files created by myself and Dr. Halls were used to publish several services using the process outlined in Appendix P.
2. I attempted to publish the raster imagery data as a faster-running image service however this process requires copying the data to the server. With the size of raster data usually being several GB, this operation would have been prohibitive.
3. It was discovered that files of type File GeoDatabase Raster (FGDBR) format required each layer to be published as a separate service in order to differentiate between layers, therefore multiple services were installed for Masonboro’s Imagery component.
4. In order for ArcGIS Online to be able to see each service, it was necessary for me to log into the server manager and set each service to “public” availability (default is private).

Create ArcGIS Online Web Map:

1. Dr. Halls worked with Esri to set up an administrator account for herself on ArcGIS Online. She was then able to send invitations to users of her designation (including myself) to join.
2. In ArcGIS Online, I created a Masonboro folder and then proceeded to create separate Web Maps to reflect each of the 4 .mxd files I originally uploaded. Appendix Q outlines the process used for Web Map construction.
3. For two layers I enabled popup windows to demonstrate query popups. I also added a push-pin and a rectangular area to one Web Map, this was a type of functionality and customization that was uniquely easy to add to Web Maps.

4. Web Maps retain the final extent you leave them in prior to saving, so I set the desired extents prior to quitting.
5. Before exiting, I set each Web Map to be publicly shared, the two options were my group (UNCW) or public.

Create JavaScript Application from Template:

1. I downloaded the previously-chosen template: “Storytelling Side Accordion – BETA” from esri.com.
2. Relevant steps were documented and modified to be used as an example and can be found in Appendix S.
3. I updated the application’s comments to better instruct students on how to complete the template.
4. The functionality to show descriptions for each map which appeared in the sample of their template wasn’t in the downloaded template (it is a beta version) so I updated it to allow students to enter this data.
5. I tested the ability to add customized widgets with mixed results:
 - a. Table of Contents: I tried adding the Table of Contents widget used for Oculina however it failed with a “Node does not exist” error. One pitfall of this widget is the lack of thorough documentation, making it incredibly difficult to diagnose and fix problems of this nature. This widget was a great option when it worked but wouldn’t be a good option for use by multiple students if it failed this easily.
 - b. Custom widget: Since most widgets perform very specific functions, it was challenging to pick one that would suit every student’s purpose. I settled on designing a simple widget that would allow students to toggle through layers. In other words, for each map “pane” a user might select, there might be several sub-layers. Clicking this button would allow the user to cycle through those layers and view/query each one individually. While testing this functionality, I discovered a quirky fact about how ArcGIS Online compiled and sent layers behind the scenes (remember they took care of these details in this setup). On queryable layers it appeared that sometimes layers were added to a

map twice in different forms: one form of the layer was an instantiation of the FeatureLayer class, an easily queried class, the other form of the layer was either an ArcGISDynamicMapLayer or an ArcGISTiledMapServiceLayer. The two different forms of the layer presented some challenges for toggling layers in this manner from the client side. An alternative approach would've been to create a greater number of WebMaps on ArcGIS Online and then toggle between those rather than trying to break down one Web Map into its multiple constituent layers.

6. Finally I altered the fields in the config.js file required to configure the map in terms of text, maps and display.
7. The Riverrun application was then placed on //server03, the GIS application web server.

3.2.5 Web Portal Update

3.2.5.1 Web Portal Tools

For the Sample Application Tools, the following tools were used:

- Microsoft© Windows™ 7 64-Bit OS
- Google© Chrome™ Version 29.0.x
- Microsoft© Internet Explorer versions 9 and 10
- Mozilla Firefox version 17.0.x
- Cisco© AnyConnect™ VPN Client
- Notepad++ Version 6.3.x

3.2.5.2 Web Portal Methodology

Web training was posted in an online format for UNCW starting in September, 2012. This timely change allowed me to complete the training remotely, which was useful since I was working on this project from Utah at the time. Following this training I was granted modification permissions for the GIS web server folder containing files related to the GIS portal. Officially UNCW requires website modifications to be made using Adobe©

Contribute™. Since the modifications to the portal required were minimal, Notepad++ was used instead for basic text modification.

The following steps were taken to update the GIS portal:

1. A back-up of the current GIS portal was made prior to alteration and saved on the network drive.
2. A copy of the current portal was then made for modification.
3. Broken project URLs were eliminated (most pointed to projects hosted on the 10.0 server)
4. URLs for projects hosted on the 9.2 server have been temporarily retained at the bottom
5. Both of the newer iterations of Oculina and Riverrun had been posted with their respective descriptions.
6. The Masonboro sample template with data had also been posted to the portal in the same format.
7. A small, clickable, linked image of each site was placed to the left of each site description. Each image was approximately 154px wide by 100px high.
8. The picture links and accompanying description each occupied their own rows in an embedded table.
9. Future updates can just add further rows to the same table following the same template.

4. RESULTS

4.1 ArcGIS Server 10.1 Install

Since the ArcGIS Server 10.1 install wasn't originally planned for in the project's conception, the resultant outcome was a system with no defined endpoint but with steadily increasing functionality (refer to Appendix K for the final system). The system began in April, 2013 as a single-machine installation but slowly became a more distributed system. Map service data was originally stored on the same server as ArcGIS Server 10.1. This was necessary as the ArcGIS Service ran using permissions from a local account and could only access local map data. Given the limited size of the local drive, space was quickly exhausted

and alternate options had to be pursued. The ArcGIS service was reconfigured to use a domain account with the proper credentials (#####@uncw.edu) and the data store was reconfigured to draw from the network drive.

Up until October, 2013, services were only visible from within the UNCW network. With ITSD's assistance, a new server was provisioned for two purposes:

- Hosting of custom JavaScript applications
- Hosting the ArcGIS Web Adaptor

Applications were then made accessible to the public through this machine using the URL http://gis.uncw.edu/arcgis/<application_name>/. Services could be accessed using the URL http://gis.uncw.edu/arcgis/rest/services/<folder_name>/<service_name>/MapServer

The remaining system components were left with default configurations as outlined in Section 3.2.1.4.

4.2 Application Migration

The deliverable defining the endpoint for application migration was: “Oculina and Riverrun confirmed to function on the new server with a similar level of functionality as permitted in the new environment” (Appendix B, Section 4.4). This requirement was met and is described over the next two sub-sections. Elements identified for potential migration are listed in Appendix D; discussions with Dr. Halls helped to determine which elements were critical for migration. The migration status of each element along with explanations for each can be found in Appendix E. Application elements that were deemed non-critical or no longer necessary were not migrated.

4.2.1 Oculina

The updated version of Oculina was posted on the GIS web server and can be viewed using the following URL: <http://gis.uncw.edu/oculina/>

The following are screenshots of the old and new versions of the Oculina applications:

Old:

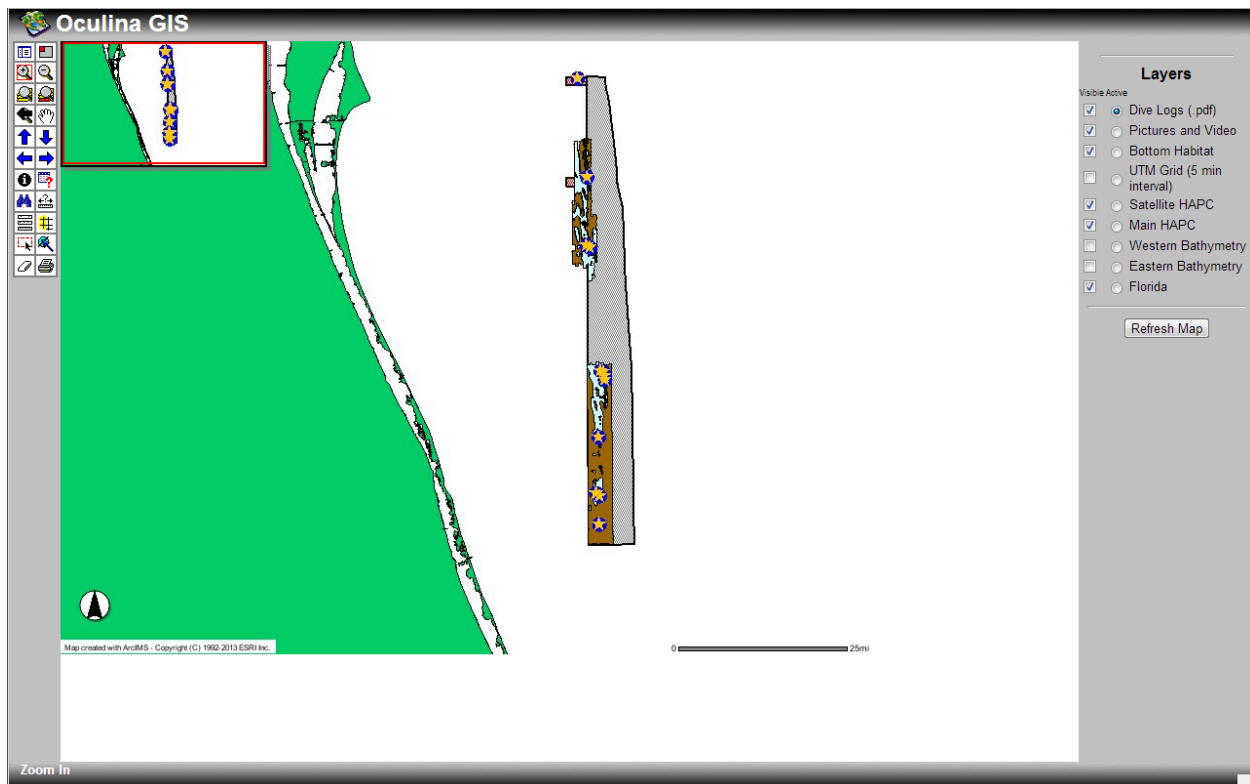


Figure 31: The original ArcIMS version of Oculina.

New:

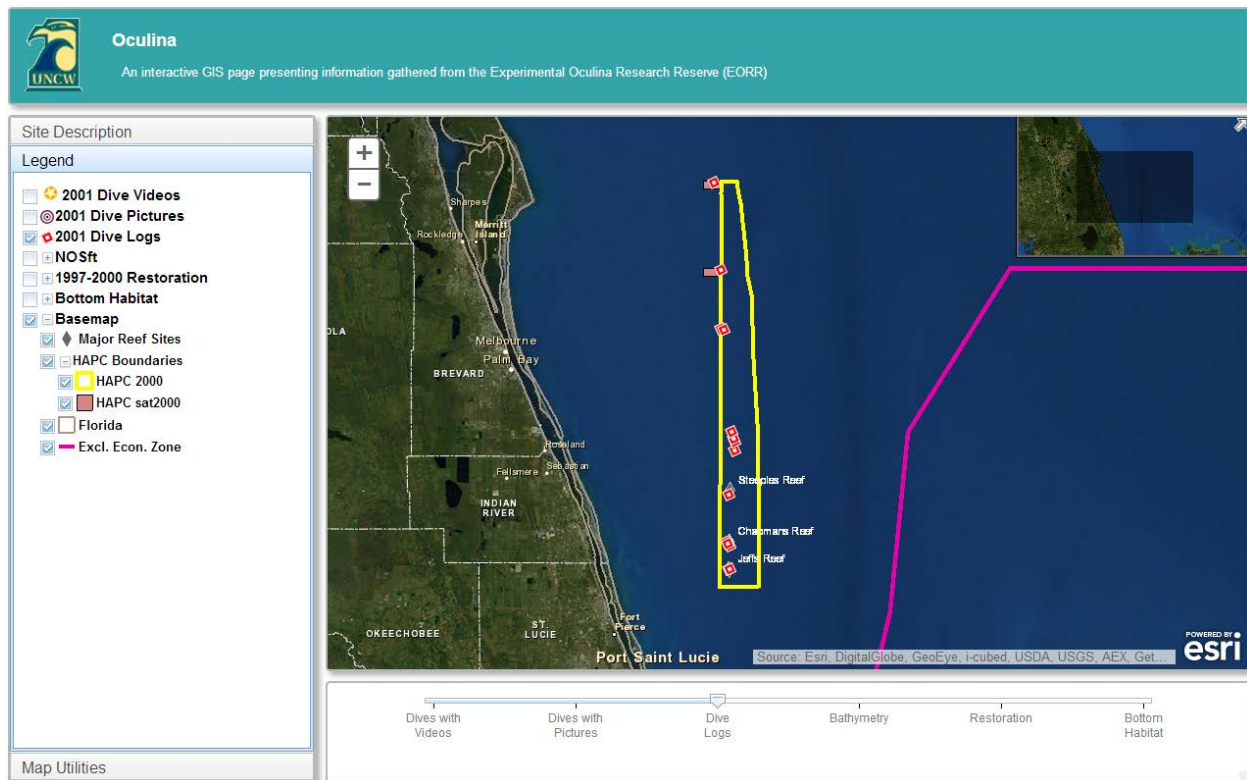


Figure 32: The newer version of Oculina hosted on the //server02 server, using services hosted on ArcGIS Server 10.1

4.2.2 Riverrun

The updated version of Oculina was posted on the GIS web server and can be viewed using the following URL: <http://gis.uncw.edu/riverrun/>

The following are screenshots of the old and new versions of the Riverrun applications:

Old:

Riverrun: An Interactive mapping application for visualizing water quality parameters in the Lower Cape Fear River. Check out the complete [Riverrun website](#).

The following are links to the interactive maps:

Overview of River Run GIS	Fish species of the Lower Cape Fear	pH
Dissolved Oxygen	Hurricanes Impacting NC	Phosphorus Total
Chlorophyll a	Nitrate & Nitrite	Salinity
Conductivity	Nitrogen Total	Temperature
Fecal Coliform	Orthophosphate	Turbidity

Figure 33: Multiple Oculina maps were accessible via link on the WebGIS Portal.

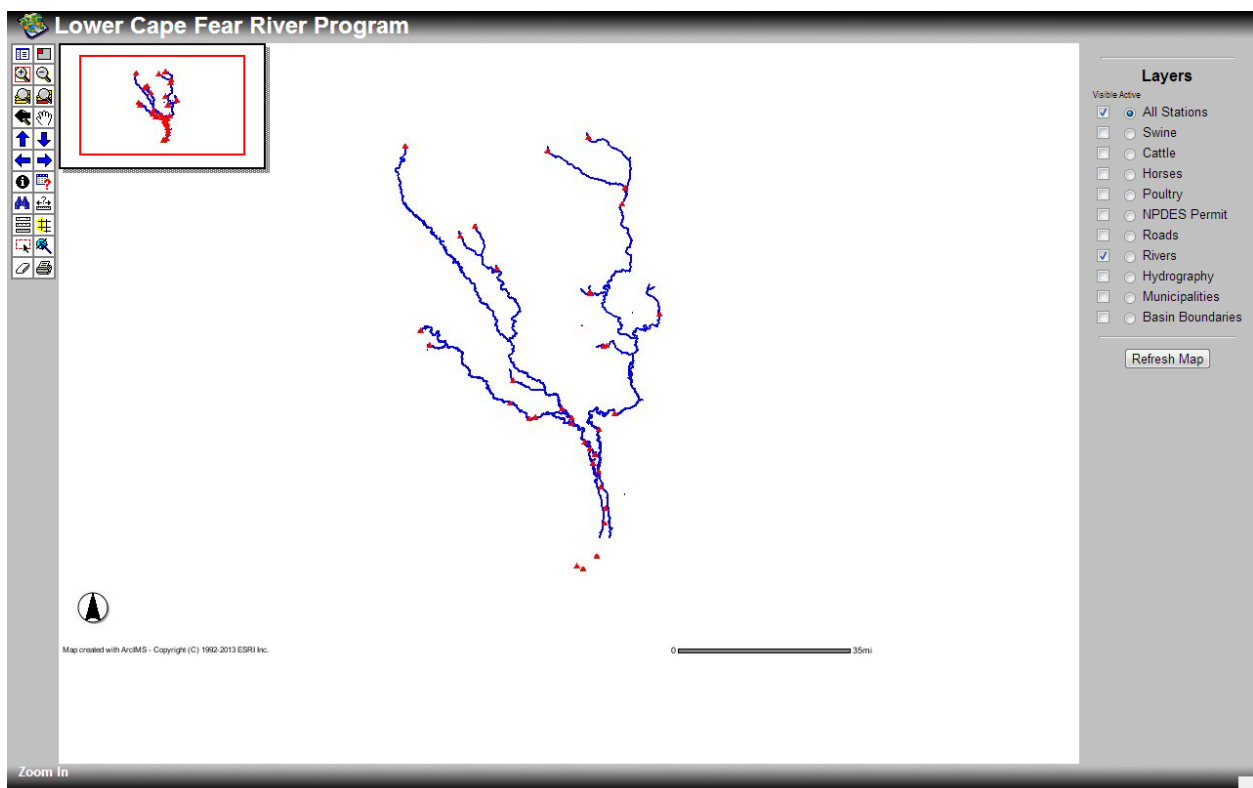


Figure 34: The original ArcIMS version of Riverrun.

New:

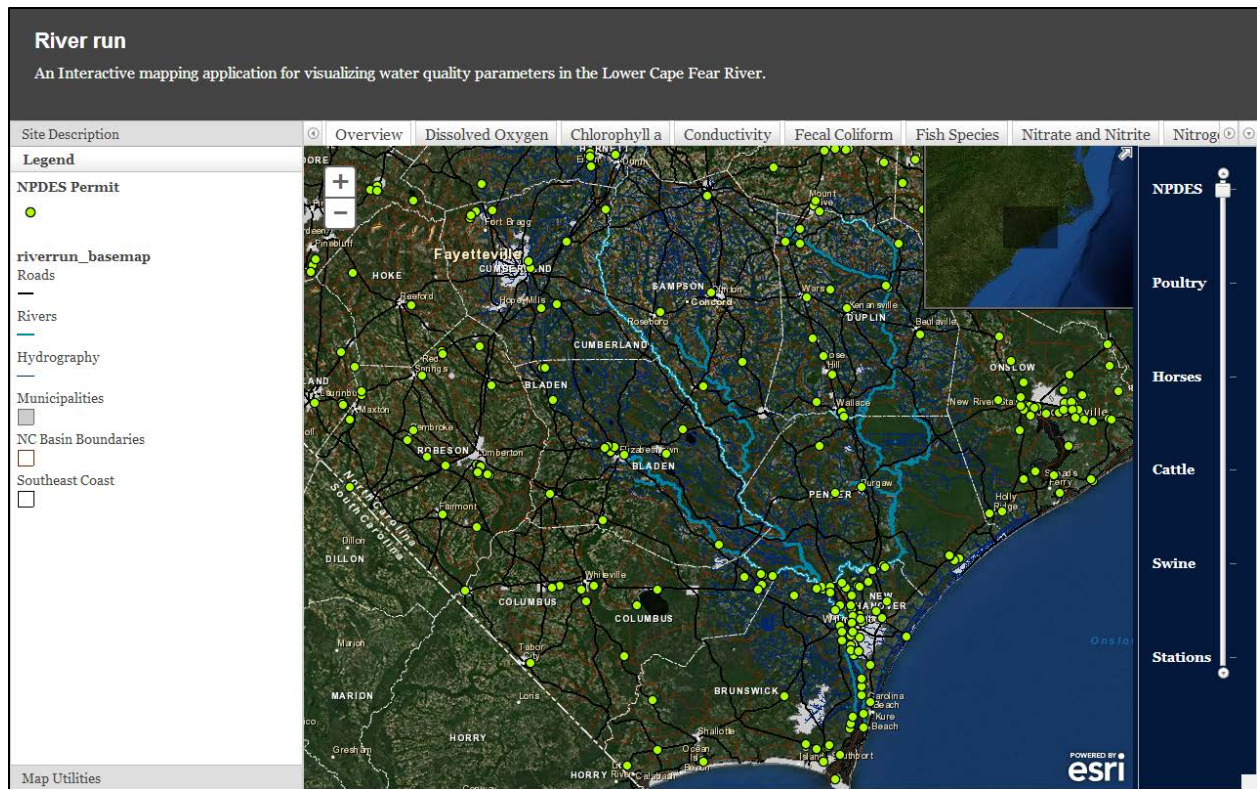


Figure 35: The newer version of Riverrun hosted on the //server02 server, using services hosted on ArcGIS Server 10.1. Unlike the previous version, all maps are available via multiple tabs in the same application.

4.3 Sample GIS Application

Sample Application completion is defined in Appendix B, Section 4.4 as follows: “A sample web application from Esri’s provided JavaScript templates will be modified to allow ‘novice developer’ code modification by UNCW GIS students for application creation. This will be accompanied by instructions for proper application creation.” The sample GIS Application was created using a modified version of the “Storytelling Side Accordion – BETA” template. The modified version this template was placed here for student access:

\\arcshare\GIServer\public\webgis\application_templates

The following figure is a screenshot of the finished template as posted on the \\server03 web server. A live version can also be found here: <http://gis.uncw.edu/masonboro/>

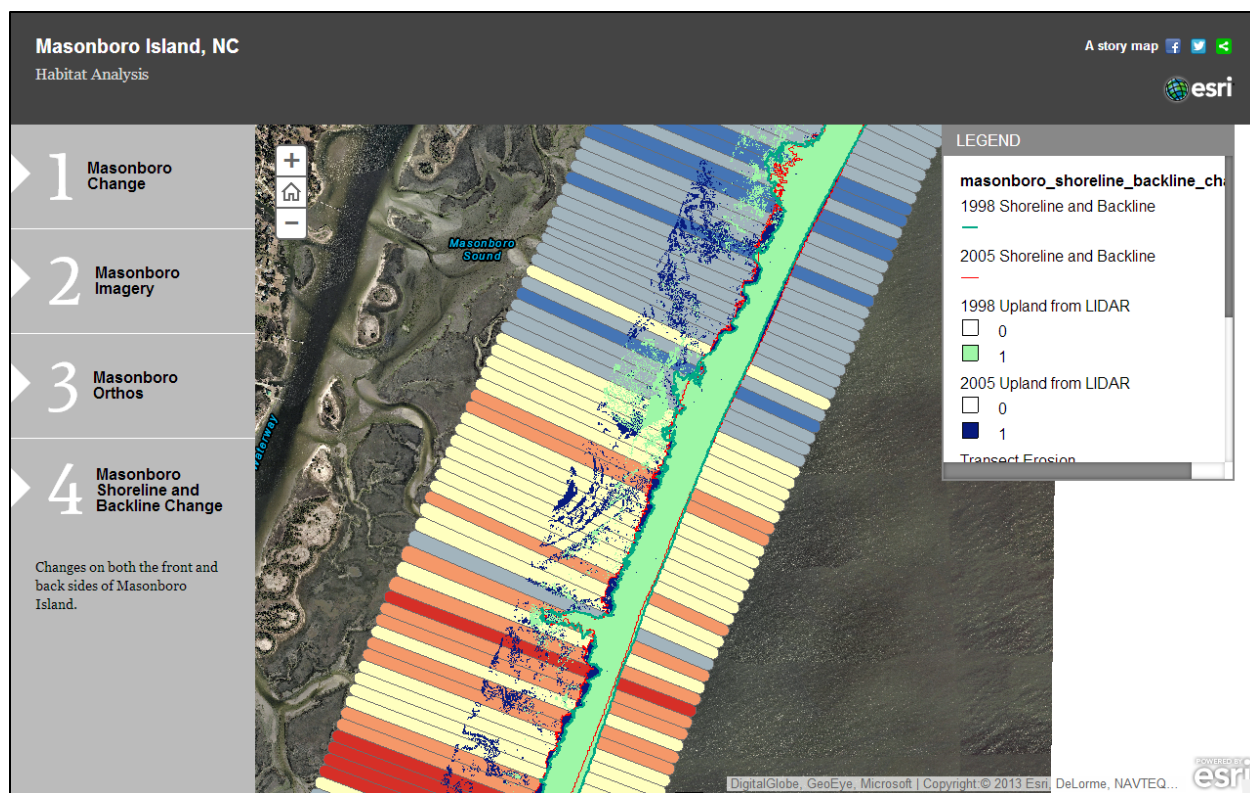


Figure 36: The Masonboro template for student application development.

Instructions creation of an App using the template or hosted on ArcGIS Online without using the template are included in Appendices Q through T. Each of the following contributing areas is covered:

- Map Service creation
- Web Map Creation
- ArcGIS Online Application Creation
- JavaScript Application Creation

4.4 Portal Update


The portal will continue to grow following completion of this project as new applications are developed and added to the list. The point of completion is defined in Appendix B as follows:

Completion of the GIS portal update was met by completion of the following three requirements:

1. (pre-requisite to website modification) Web Content Manager and Editor Training completed. This training was mandated by UNCW for staff, faculty and students who wish to modify websites representing either the university or one of its constituent colleges
2. Successful application migration and subsequent posting of publicly accessible URLs with accompanying descriptions.
3. Uploaded sample template

Though this was a small part of the project, there was some room for improvement. The idea for using map thumbnails was lifted from Esri's JavaScript templates website.

Old version of the portal:


UNIVERSITY of NORTH CAROLINA WILMINGTON

[Email](#) | [Directory](#) | [mySeaport](#) | [Quicklinks](#) ▼

Geographic Information Sciences

[Current Events](#)
[Curriculum](#)
[Faculty & Staff](#)
[Research](#)
[Resources](#)
[Web GIS Portal](#)

Web GIS Portal

The Spatial Analysis Lab in the Dept. of Geography and Geology has created Internet Mapping websites since 2000 and has recently upgraded to ArcServer 10 for hosting several GIS projects. You can link to these Web-GIS services using the list below.

[A GIS Based Multi Criteria Evaluation of Hurricane Risk in Coastal Counties](#) by Sean Burke, UNCW Graduate GIS Certificate Program, Spring 2012
 The purpose of this GIS is to identify specific coastal counties and regions in the Atlantic hurricane basin that are at a higher risk to the damaging effects of hurricanes.

[Land Use and Water Quality Characteristics in New Hanover County Tidal Creeks](#) by Kaitlyn Costin, UNCW Marine Science Masters Program and GIS Certificate Program, Spring 2012
 The purpose for this project was to determine if there is a spatial relationship between various types of land use changes and changes in several water quality variables.

[Habitat Suitability Index Model: Red Cockaded Woodpeckers, Brunswick, NC Corridor](#) by Cory Darnell, UNCW Marine Science Masters Program and GIS Certificate Program, Spring 2012
 The objective of this study was to identify suitable RCW habitats and potential corridors that could connect the Green Swamp and Boiling Springs Lakes Preserves in Brunswick County, NC.

[The Application of LIDAR Data to Classify and Analyze Change of Barrier Island Morphologic Features](#) by Timothy Moss, UNCW Marine Science Masters Program, Spring 2012
 The objectives of this study were to classify multiple barrier island features and to quantify the evolution of each feature over time by calculating the change in volume.

[Spatial Analysis of Sea Turtle Nesting Frequency Along Topsail Island and Masonboro Island](#), North Carolina by Ben Sciance, UNCW Graduate GIS Certificate Program, Spring 2012
 The purpose of this study was to investigate the influence of environmental and anthropogenic variables on the frequency of sea turtle nesting on two North Carolina barrier islands.

Older ArcIMS 9.3 websites are still operational, but are undergoing a transition to the new ArcServer platform. While this

Figure 37: The portal with the older application links, most of these connected to the intermittently functional version 10.0 server.

New version of the portal:

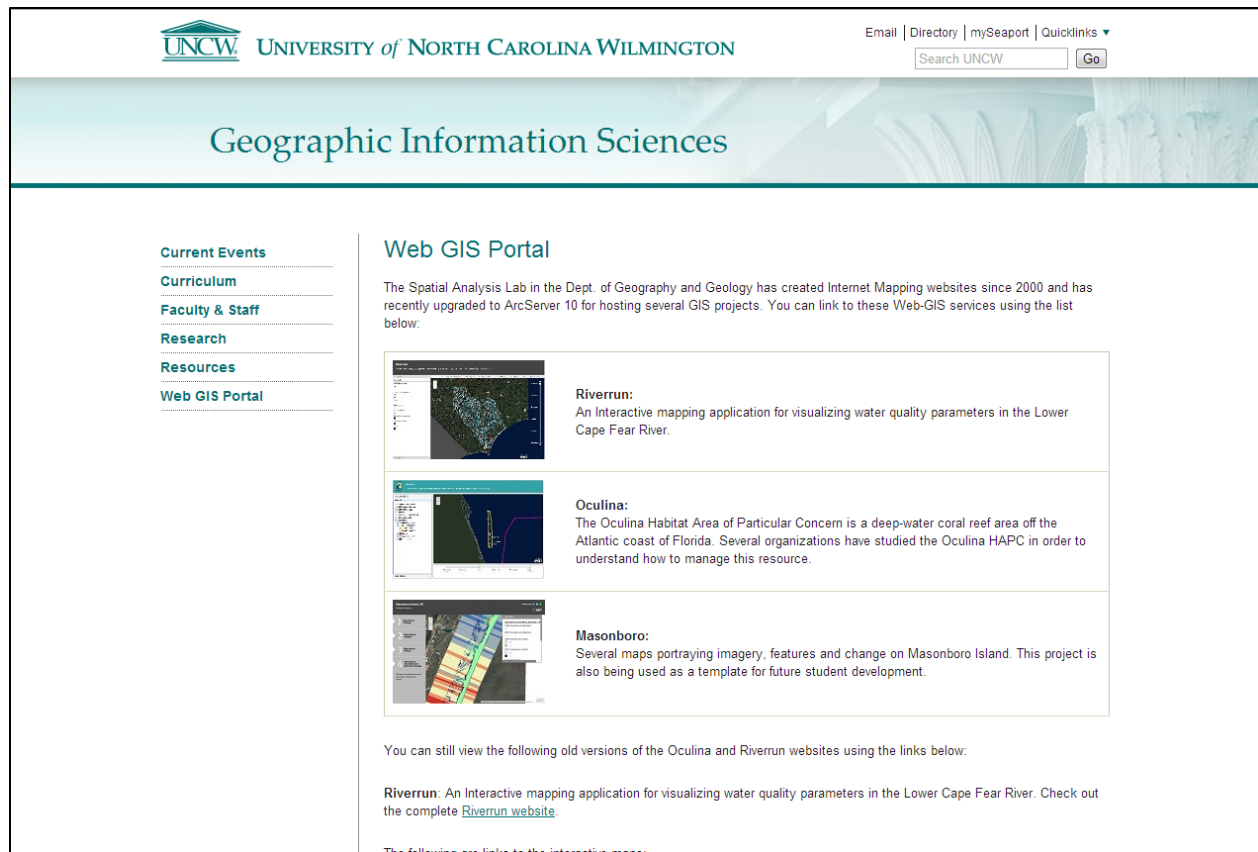


Figure 38: New portal design with thumbnails for map applications. The older version 9.2 applications were retained at this point but may be removed following this project's completion.

The new portal can be accessed here: <http://www.uncw.edu/gis/webgisportal.htm>

4.5 System Design

The following requirements were outlined in Appendix B, Section 4.4 for completion of the analysis and design portion of this project:

The following object oriented analysis & design concepts will be delivered for a complete conceptual view of the web server:

- Use case models
- Business process models for service and application creation on each of the three servers
- System component analysis, wireframes for application design

The elements of this list changed considerably from the original Project Charter to the final revised Project Charter, the following table explains why:

Original Proposed Project Charter Requirement	Final Project Charter Requirement	Change Justification
Use case models	Use case models	No change, these were retained
Class, Sequence, Activity and State Diagrams	Business process models for service and application creation on each of the three servers	A 'service-oriented architecture design' approach was used. This allows for some flexibility in the manner in which charts and diagrams were used. Since Dr. Halls and the students (the primary audience) will not be heavily leveraging the API, the Business Process Model was chosen as the most useful diagramming method for comparisons of prior and future service and application development.
Software Requirements Specification (SRS)	System component analysis, Wireframes for application design	Given further investigation and the developer's professional experience, it was decided that an SRS would be better suited for large-scale application development. A targeted approach of analyzing ArcGIS Server system components and developing wireframes for the prospective applications was chosen as a more useful approach.

Figure 39: Justification for requirements change in revised charter.

Actor Diagrams for Service and Application creation can be found in Appendix H. The process for creating services and applications changed considerably from ArcGIS Server 9.2 to 10.0 and again from 10.0 to 10.1. Since this project's Analysis & Design spanned service and application development across 3 Server versions, business process models were created for service and application creation on each, they can be found in Appendices L, M and N.

The analysis of the system components was covered previously in Section 3.2.1, the design, drawn from the results of the analysis, is covered below.

The results of the tests performed by Sakowicz & Pizzi (2012) covered in Section 3 combined with in-house tests on both the memory and CPU load for current service operations, indicated that improvements should focus first on the # of CPUs on the ArcGIS Server. The VM had two CPUs. As regular demand increases, it could be prudent to increase the number incrementally in much the same way as was done in Sakowicz & Pizzi's tests. Tests indicate that raster data processing on the server consumed large amounts of memory which is unsurprising given that the currently installed amount is the minimum defined by Esri.

In-house tests of read/write speeds on both the NAS and local VM storage yielded the results found in the figure below.

		Local (C:\)								Network Attached Storage (\\arcshare)							
		Seq		512k		4k		4k QD32		Seq		512k		4k		4k QD32	
Date	Time	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write
10/11/2013	12pm	67.2	89.3	29	43	0.32	0.66	1.18	0.91	102	150	84.9	141	9.9	8.81	21.3	19.1
	6pm	71.8	89.1	31.1	51.1	0.4	0.81	4.67	2.47	110	193	109	181	11	10.2	41.1	27.5
10/14/2013	4pm	68.4	89.3	27.5	42.7	0.33	0.73	1.26	1.64	132	187	136	183	13.7	11.9	37.7	28.3
10/15/2013	3:30pm	67.3	85.5	29.5	44.4	0.34	0.71	1.29	1.08	124	188	132	176	13.3	12.4	42.6	27.9
10/16/2013	1:30pm	59.4	91.6	26.1	44.1	0.34	0.68	1.06	0.88	134	198	134	186	13.1	11.7	47.2	27
	4:30pm	69.3	107	27.8	49.6	0.39	0.69	1.43	1.23	136	193	130	178	13	12.1	45	24.8
	Average:	67.2	92	28.5	45.8	0.35	0.71	1.81	1.37	123	185	121	174	12.3	11.2	39.2	25.7

Figure 40: Results (in MB/s) obtained testing read/write speeds from VM hosting ArcGIS 10.1 to the local drive for NAS. In every test the NAS outperformed the local drive (larger #s are better) (a larger version of this chart may be found in Appendix O).

Read/write speeds for sequential block as well as random blocks of varying sizes universally performed better with the NAS when compared to the VM local drive. These tests were conducted across a range of different times however performance remained relatively consistent. From a management and speed standpoint, there appeared to be no reason to discontinue storage of application data on the NAS.

The analysis and installation of ArcGIS 10.1 were done in parallel, so many of the recommended system improvements were implemented along the way by me, Dr. Halls

and/or ITSD. The final result of these configurations is reflected in the system diagram in Appendix K. The following Analysis & Design suggestions were recommended and subsequently implemented:

- Installation of the ArcGIS Web Adaptor
 - Firewall exception for server to allow public viewing of applications
 - Installation of IIS 7 Server for hosting of Web Adaptor and JavaScript Applications
- Provisioning of domain user name for ArcGIS Service operation to allow access to NAS resources
- Setup of ArcGIS.com accounts for Web Map Creation and facilitation of Application Development

Further component improvements could include the following:

- Dr. Halls showed a great deal of interest in posting raster data. The ability to host some common Image Map Services that many applications would use could be a useful feature. This would require a substantial increase in VM Hard Drive space. Dr. Halls would want to work closely with ITSD to determine a reasonable plan for this.
- Use of ArcSDE (Esri's spatial database engine) is included with the current site license. Implementation of ArcSDE would include the following benefits:
 - Speedier service response times
 - Elimination of the Server bug inherent to ArcGIS Server 10.1 (See Section 3.2.3.4)
 - Quicker Raster services (a good option if expanding the VM hard drive to accommodate Image Map Services isn't an option)
 - Industry-standard instruction. Many large business work with ArcSDE since it's capable of building on top of many common business databases.
 - More practical use of UNCW's site license
 - Centralized repository for data
 - Less data redundancy
 - Permission of simultaneous read/write of data

- There were several further improvements that could be made to the system however they're secondary to those that had been recommended already (and in some cases already implemented). Nevertheless, here are a couple more optional system improvement to consider later if needed:
 - Increase number of GIS Servers. This was shown to be secondary to increasing the number of CPUs however it should be considered if the number of CPUs on a single VM gets maxed out.
 - Design applications to enable client caching of data (advanced, for JavaScript development)
 - For commonly used base layers where there's no requirement to turn individual layer visibility on or off, consider creating tiled map layers

5. CONCLUSIONS, LIMITATIONS & RECOMMENDATIONS

5.1 Application Development

Esri's Software Development landscape was at times confusing to navigate. As a beginning user (even one with coding experience), it wasn't immediately clear how widgets and templates fit together. It also wasn't clear where applications could or couldn't be hosted. Here are several facts that were difficult to determine relating to application development using Esri's JavaScript API:

1. **Templates are primarily designed for interface with Web Maps on ArcGIS.com.** This means they're hosted on a web server of your choosing (usually your own); these applications would then communicate with ArcGIS Online to gather map information. If you really understand the templates inner workings, you can potentially rework them to load layers directly from services using the API however this is not always an easy task depending on the complexity of the template. A user would be best off "hollowing out" a very simple template and effectively starting out from scratch or just starting from scratch using Dojo UI elements if interested in creating a highly customized ArcGIS JavaScript application.

2. **The widgets that Esri highlighted on their JavaScript API website are mostly useful for the highly customized applications I've just described.**

They can be implemented in templates that interface with ArcGIS Online however they likely won't follow the same patterns you're seeing in the widget examples. For example, say you have a widget that performs an action on individual sub-layers. An example might show the user creating each layer and then altering its properties. When you interface with ArcGIS Online from an application, you still end up with a map object which contains layers. In this circumstance however, you need to pull the layers out and to determine properties or make changes. This gets trickier as more layer types get mixed together in a single map. Further challenges may ensue when ArcGIS Online adds the same layers using multiple different layer classes to one map.

3. **If at all possible, tailor your task to fit a template, not the other way around.** In much the same way that Enterprise Solutions can provide generic solutions that benefit a broad array of businesses, Esri's JavaScript templates provide general solutions that are quick and easy to implement. If your data doesn't match the template, at least consider either using a more appropriate template or altering how your data is portrayed in order to fit the template prior to building a completely customized application. To put some numbers behind this, here was my approximate development time for the three applications:

Application	Development Time
Oculina (customized JavaScript)	~500 Hours
Riverrun (customized JavaScript)	~180 Hours
Masonboro (template JavaScript)	~20 Hours

Figure 41: Approximate application development time.

It should be noted that much of the time devoted to Oculina went to "discovery" of how to work with the ArcGIS JavaScript API, Dojo, JQuery, and to a lesser degree, HTML. Likewise it should be noted that much time was saved on Riverrun given the use of Oculina as a rough template.

4. **If possible, consult with an industry professional prior to estimating or agreeing to development tasks.** It can very difficult to determine the level of complexity of developing on a new API without consulting someone who's familiar with the inherent challenges first.

5.2 Analysis & Design

It's challenging both from the developer and the client's perspectives not to want to implement changes to the system that will be of clear benefit. While it may be desirable to make system improvements, getting caught up on this requires time, energy, meetings and further documentation. It's important to make the distinction between Analysis & Design and System Modification. While one may lead to the other, Analysis & Design and System Modification should remain separate. With that said, sometime some improvements may be required in order for other aspects of a project to work. For instance, application development would've been hampered had I not gotten involved with provisioning of a web server and installation of the ArcGIS Web Adaptor.

5.3 Limitations

Alluding to the point in Section 5.1 regarding consulting with a professional prior to work engagement, this action might also help to pre-establish any known limitations. It's far easier to set realistic expectations and then exceed them where possible vs. setting high but poorly researched expectations initially and then experiencing some level of disappointment when these expectations can't be met.

There were several limitations on this project including working remotely, significant changes in the server architecture and changes to the API and application design. Rather than taking a literal approach to application migration (feature for feature transfer), both I as the developer and the client, might have been better served by determining the best ways to use the new API and architecture to design an application that tells a better story.

6. FUTURE WORK

Service migrations:

Though outside of the scope of this project, there were several valuable services that were developed for the ArcGIS 10.0 Server which could be easily ported to the new Server

Further portal updates:

The portal may require further updates to reflect new student applications. Should there be any; the portal will be updated appropriately.

Masonboro Template adjustment:

Though the Masonboro template was functionally complete, it was capable of showing far more data more user-friendly ways and may be adjusted to reflect this.

Campus IE Explorer Issue

ITSD was informed of a server-browser issue that resulted in a failure to load the application on campus workstations using IE 10. This issue could be mitigated by using the IP address of the service in place of the URL or by using the latest versions of Chrome or Firefox. A long-term fix should be implemented.

Miscellaneous Documentation

I'm happy to work with Dr. Halls to augment or provide further instructions on some of the processes as needed.

7. REFERENCES

"ArcGIS Help 10.1." ArcGIS Resource Center. N.p., n.d. Web. 15 Apr. 2013.

"ARK | Intel® Xeon® Processor E5640 (12M Cache, 2.66 GHz, 5.86 GT/s Intel® QPI)." ARK | Your Source for Intel® Product Information. Intel, n.d. Web. 9 Oct. 2013.
<<http://ark.intel.com/products/47923/>>.

Bhattacharjee, Biswapati; Baleja, John; Perez, Ricardo; Catanzano, Jon. " Esri® ArcGIS® 10.1 for Server

Bieberstein, Norbert. Executing SOA a practical guide for the service-oriented architect. Upper Saddle River, N.J: IBM Press/Pearson plc, 2008. Print.

Brimicombe, Allan. GIS, environmental modeling and engineering. 2nd ed. Boca Raton: CRC Press/Taylor & Francis Group, 2010. Print.

Carnes, Ray. "Difference between ArcIMS and ArcGIS Server ." HOME – ESRI Discussion Forums. Esri, 2 Dec. 2004. Web. 18 Jan. 2013.
<<http://forums.esri.com/Thread.asp?c=158&f=1696&t=143695>>.

"Configuring a distributed installation of ArcGIS Server." Help for Previous Versions | ArcGIS Resources. Esri, n.d. Web. 27 Sept. 2013. <
http://webhelp.esri.com/arcgisserver/9.2/dotnet/manager/administration/distributed_install.htm#add_soc_machines>

Dempsey, Caitlin. "Free Map Servers – GIS Lounge." GIS Lounge – GIS, Maps, and Cartography. GIS Lounge, 1 July 2010. Web. 10 Oct. 2012. <<http://www.gislounge.com/free-map-servers/>>.

"Desktop GIS." MapInfo.com RSS. N.p., n.d. Web. 2 Oct. 2013.
<<http://www.mapinfo.com/products/desktop/>>.

"Esri ArcGIS 10.2 for Server on VMware® vSphere® - Technical Considerations Guide." Esri Support. July, 2013. Esri. Web. 02 Oct. 2013. <
http://downloads.esri.com/support/whitepapers/other/_W27387_Esri_ArcGIS-VMware_Deployment_Guide_FINAL_080513.pdf>.

"Esri International User Conference." Esri Events | GIS User Conferences – Seminars – User Groups. Esri, n.d. Web. 18 Jan. 2013.
<<http://events.esri.com/uc/QandA/index.cfm?fuseaction=answer&conferenceId=2a8e2713-1422-2418-7f20bb7c186b5b83&questionId=2579>>.

Farkas, Keith I.; Narayanaswami, Chandra; Nieh, Jason; "Guest Editors' Introduction: Virtual Machines," Pervasive Computing, IEEE , vol.8, no.4, pp.6-7, Oct.-Dec. 2009

"GEOSPATIAL TECHNOLOGY." Employment & Training Administration (ETA) – U.S. Department of Labor. United States Department of Labor, n.d. Web. 13 Feb. 2013.
<www.doleta.gov/BRG/pdf/Geospatial.pdf>.

“Geospatially Powered Solutions.” Intergraph News. N.p., n.d. Web. 2 Oct. 2013.
<<http://www.intergraph.com/sgi/default.aspx>>.

“High Growth Industry Profile – Geospatial Technology.” Employment & Training Administration (ETA) – U.S. Department of Labor. United States Department of Labor, 8 Mar. 2010. Web. 10 Feb. 2013. <http://www.doleta.gov/brg/Indprof/geospatial_profile.cfm>.

“IDEF0.” Welcome to IDEF.com. Knowledge Based Systems, Inc., n.d. Web. 3 Oct. 2013.
<<http://www.idef.com/idef0.htm>>.

“Installing ArcGIS for Server.” ArcGIS Resource Center. Esri, 13 Dec. 2012. Web. 3 May 2013.

“Installing ArcGIS for Server.” ArcGIS Resource Center. N.p., 13 Dec. 2012. Web. 27 Sept. 2013. <http://resources.arcgis.com/en/help/install-guides/arcgis-server/10.1/index.html#/Deployment_scenarios/01nm00000021000000/>.

Johnson, Shawana P.; “Marlow, Susan; GIS Impact: Department of Labor Declares Geospatial Industry High Growth.” Esri proceedings. Esri. 12 Feb. 2013.
<<http://proceedings.esri.com/library/userconf/proc05/papers/pap1432.pdf>>.

Klinkenberg, Brian. “UNIT 23 – HISTORY OF GIS.” UBC Department of Geography. University of British Columbia Department of Geography, n.d. Web. 2 Mar. 2013. <<http://www.geog.ubc.ca/courses/klink/gis.notes/ncgia/u23.html#SEC23.1>>.

McKenna, Frank. “THIN-CLIENT VS FAT-CLIENT COMPUTING.” THIN-CLIENT VS FAT-CLIENT COMPUTING. KnowledgeOne Corp., 1 Jan. 2002. Web. 14 Oct. 2013.
<<http://www.knowledgeonecorp.com/news/pdfs/Thin%20client%20vs%20Fat%20client%20Computing.pdf>>.

“Model-based GIS and mapping software.” AutoCAD Map 3D. N.p., n.d. Web. 2 Oct. 2013.
<<http://www.autodesk.com/products/autodesk-autocad-map-3d/overview>>.

“Moore’s Law.” Merriam-Webster.com. Merriam-Webster, n.d. Web. 17 Nov. 2013.
<[http://www.merriam-webster.com/dictionary/Moore’s law](http://www.merriam-webster.com/dictionary/Moore's%20law)>.

“Planning an ArcGIS Server site configuration.” EDN: ESRI Developer Network. Esri, n.d. Web. 10 Dec. 2011.

<http://edndoc.esri.com/arcobjects/9.0/arcgisserver/helppages/server_site_config.htm>.

Sacowicz, Andrew; Pizzi, Frank. “ArcGIS for Server Performance and Scalability—Testing Methodologies.” March 26-29, 2012, Palm Springs, CA. Esri, 2012. Print.

Sauter, Vicki. “OBJECT-ORIENTED ANALYSIS.” University of Missouri-St. Louis.

University of Missouri-St. Louis, 29 Nov. 1999. Web. 3 Oct. 2013. <

<http://www.umsl.edu/~sauterv/analysis/ooa.html>>.

“Server Software Performance.” Wiki.gis.com. N.p., 10 Sept. 2013. Web. 16 Sept. 2013. <

http://www.wiki.gis.com/wiki/index.php/Server_Software_Performance>.

Spewak, S. H., and Michael Tiemann. “Updating the Enterprise Architecture Planning Model.” *Journal of Enterprise Architecture* 2.2 (2006): 11-19.

“System Requirements.” ArcGIS Resource Center. N.p., 7 Oct. 2013. Web. 12 Oct. 2013.

<<http://resources.arcgis.com/en/help/system-requirements/10.1/index.html#/0151000000720000>>

Wikipedia contributors. “Object-oriented analysis and design.” Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 16 Sep. 2013. Web. 3 Oct. 2013.

Wikipedia contributors. “Service-oriented modeling.” Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 29 Jul. 2013. Web. 3 Oct. 2013.

Wikipedia contributors. “Structured Analysis and Design Technique.” Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 3 Mar. 2013. Web. 3 Oct. 2013.

Worboys, Michael, and Matt Duckham. *GIS: a computing perspective*. 2nd ed. Boca Raton, Fla.: CRC Press, 2004. Print.

Zimmerman, Olaf, Pal Krogdahl, and Clive Gee. “Elements of Service-Oriented Analysis and Design.” IBM – United States. N.p., 6 Feb. 2004. Web. 3 Oct. 2013.

<<http://www.ibm.com/developerworks/webservices/library/ws-soad1/>>.

**APPENDIX A: PROJECT CHARTER AS PRESENTED TO THE COMMITTEE ON
NOVEMBER 9, 2012**

Project Name: Analysis, Design and Implementation of ArgGIS Server and Web Portal for UNCW's Department of Geography and Geology

Department: The Information Systems & Operations Management Department and the Computer Science Department at the University of North Carolina Wilmington

Focus Area: Geographic Information Systems

Product/Process: Server-Based Deployment of ArcGIS Server 10.0

Prepared By

Document Owner(s)	Project/Organization Role
Aaron Caldwell	Project Owner
Dr. Joanne Halls	Project Committee Member, Client
Dr. Douglas Kline	Project Committee Leader
Dr. Ron Vetter	Project Committee Member

Dr. Bryan Reinicke	Project Committee Member
--------------------	--------------------------

Project Charter Version Control

Version	Date	Author	Change Description
A	July 24, 2011	Aaron Caldwell	Document Created
B	August 1, 2011	Aaron Caldwell	Project Charter Sections Added
C	August 7, 2011	Aaron Caldwell	Project Charter Sections Populated
D	October 2, 2011	Aaron Caldwell	Complete revision for compliance with researched guidelines and input from Dr. Douglas Kline
E	September 16, 2012	Aaron Caldwell	Updated to better reflect client's requirements

Table of Contents

1. PROJECT PURPOSE	3
--------------------	---

2. PROJECT EXECUTIVE SUMMARY	3
3. PROJECT OVERVIEW	3
4. PROJECT SCOPE	3
4.1 Goals and Objectives	3
4.2 Departmental Statements of Work (SOW)	3
4.3 Key Stakeholders	3
4.4 Project Deliverables	3
4.5 Deliverables Out of Scope	3
4.6 Project Estimated Costs & Duration	3
5. PROJECT CONDITIONS	3
5.1 Project Assumptions	3
5.2 Project Issues	3
5.3 Project Risks	3
5.4 Project Constraints	3
6. PROJECT STRUCTURE APPROACH	3
7. PROJECT TEAM ORGANIZATIONS PLANS	3

1. Project Purpose

The Project Owner, with the support of the Project Committee, will perform a migration of web mapping applications from the previous ArcGIS map server to the new ArcGIS map server, an update to the current web portal to reflect migrated map applications and an analysis & design of the current map server for the Department of Geography and Geology at the University of North Carolina Wilmington (UNCW).

2. PROJECT EXECUTIVE SUMMARY

- **Project Goals:** Analyze current setup of ArcGIS Server, update GIS Web Portal and migrated map applications to new server for UNCW Department of Geography and Geology
- **Objectives:** To better utilize UNCW's ESRI software site license and provide a better platform for GIS instruction.
- **Scope:** To deliver a convenient means of accessing ArcGIS Server in a high quality format that meets project requirements within the agreed upon timeframe.
- **Assumptions:** The required infrastructure has been allocated including space on the server, ESRI ArcGIS Server software and ITSD resources for future maintenance/upkeep.
- **Risks:** Server latency due to excessive traffic.
- **Costs:** This project is being conducted by the project owner in partial fulfillment of the requirement for M.S. in Computer Science and Information Systems at UNCW. All software resources have already been allocated (see Assumptions). As such, there will be no inherent cost for this project apart from time commitments by project committee members, associated departments and the UNCW Information Technology and Services Division (ITSD).
- **Timeline:** All deliverables agreed upon by the project committee shall be met by December, 2012.

- Approach: Project Owner shall conduct bi-weekly meetings with the Project Committee. The project Owner will be responsible for executing and tracking all work agreed upon by the Project Committee. The Project Committee shall communicate via email and telephone and teleconference as necessary.

Organization:

- Aaron Caldwell (Project Owner)
 - Dr. Joanne Halls (Project Committee Member, Client)
 - Dr. Douglas Kline (Project Committee Leader)
 - Dr. Ron Vetter (Project Committee Member)
 - Dr. Bryan Reinicke (Project Committee Member)
-

3. PROJECT RATIONALE

The University of North Carolina Wilmington Geography Department purchased a software package from ESRI for Geographic Information Systems instruction. The software package site license includes software for individual workstations as well as a server component for deployment of Internet-accessible customized maps. Currently the department utilizes the workstation component of this new software package alongside both the new server and the previous server. Migrating previously built map applications from the old server to the new server and performing an analysis of the new server system will allow the UNCW Geography Department to better utilize the ESRI Software site license and provide a more consistent platform for GIS course instruction.

4. PROJECT SCOPE

4.1 Goals and Objectives

Goal(s)	Objectives
Migrate custom map applications from old server to new server	<ol style="list-style-type: none">1. Determine failure mode for map applications inability to run on new server2. Implement appropriate code modifications for operation on new server3. Publish updated, functioning map applications on new server and portal
Update ArcGIS Server Web Portal	Modify/update web portal to reflect migrated map applications
Perform an Analysis and Design of the new ArcGIS Server	Analyze: <ol style="list-style-type: none">1. GIS Server- hosts GIS resources and exposes them to client applications.1. Web Server- hosts web applications and services using resources on the server.2. Data Server- resources published on server.

4.2 Departmental Statements of Work (SOW)

Departmental SOW	Owner/Prime	Due Date/Sequence
Research and Analysis	Project Owner	July 01, 2011 – November 15, 2012

Configuration and Implementation	Project Owner	September 10, 2012 – November 15, 2012
Advising Resource	Project Committee	August 01, 2011 – December 3, 2012
Project Defense	Project Owner	December 3, 2012

4.3 Key Stakeholders

Stakeholder	Impact to and Participation of Organization
University of North Carolina Wilmington (UNCW)	Host resources physical resources and site license for ArcGIS Server
UNCW Information Technology and Security Division (ITSD)	Site contact for server resources, configuration questions and deployment. ITSD will act as a technical resource for the ArcGIS Server implementation following the project's completion.
UNCW Department of Computer Science	Faculty and staff will be consulted for server implementation expertise.
UNCW Department of Information Systems and Operations Management	Faculty and staff will be consulted for server implementation expertise.

UNCW Department of Geography and Geology	Faculty and Staff will be the primary administrators and users of ArcGIS Server Software. Department members will be consulted for input regarding initial configuration.
--	---

4.4 Project Deliverables

Milestone	Deliverable
Web application migration	All web applications that were confirmed to have functioned on the previous web server are confirmed to function on the new server
Web portal update	Links/descriptions of web applications posted to UNCW's GIS portal
Analysis & design of web server	<p>The following object oriented analysis & design concepts will be delivered for a complete conceptual view of the web server:</p> <ul style="list-style-type: none"> • Use case models • Class, Sequence, Activity and State Diagrams • Software Requirements Specification (SRS)

4.5 Project Estimated Duration

Milestone	Date	Deliverable	Confidence
-----------	------	-------------	------------

	Estimate		Level
Project Definition	September 28, 2012	<ul style="list-style-type: none"> • Define project goals • Determine configuration options • Estimate required resources 	Medium
Web application migration	October 22, 2012	<ul style="list-style-type: none"> • All web applications confirmed to have functioned on the previous web server confirmed to function on new server 	Medium
Web portal update	November 3, 2012	<ul style="list-style-type: none"> • Links/descriptions of web applications posted to UNCW's GIS portal 	Medium
Analysis & design of web server	November 24, 2012	<ul style="list-style-type: none"> • Use case models • Class, Sequence, Activity and State Diagrams • Software Requirements Specification (SRS) 	Medium
Project completion	December 3, 2012	<ul style="list-style-type: none"> • Determine that project meets stakeholder requirements • Project defense 	Medium

4.6 Deliverables Out of Scope

- Other data or documentation linked or made accessible via web portal.

5. PROJECT CONDITIONS

5.1 Project Assumptions

- UNCW ITSD is currently in possession of an ESRI site license with rights for operation of ArcGIS Server Enterprise Edition as previously confirmed.
- UNCW Department of Geography and Geology and UNCW ITSD will handle user authentication for web portal and ArcGIS Server use.
- UNCW ITSD will provide ArcGIS Server support following project completion.

5.2 Project Risks

#	Risk Area	Likelihood	Risk Owner	Project Impact-Mitigation Plan
1	Server performance latency	Low	Aaron Caldwell/ ITSD	Analysis and Design could reveal disadvantages for single machine deployment. Server latency could be mitigated by placing the Server Object Container and Server Object Manager on different machines. Project owner will advise ITSD should this situation arise.
2	ITSD Assistance	High	Aaron Caldwell/ ITSD	The Project Committee has the physical presence for face-to-face communication as well as the leverage within the UNCW system to assist with ITSD as necessary.
3	Performing Project Research and Work	High	Aaron Caldwell	Any research or operations that can't be performed remotely will be accomplished through a combination of the following in

	at a Distance			descending order of frequency: <ol style="list-style-type: none"> 1. Periodic visits to the UNCW campus 2. Assistance from ITSD 3. Assistance from Project Committee
4	Awaiting online availability of UNCW's Web Content Manager and Editor Training	Medium	Aaron Caldwell	In order to make changes to a UNCW departmental web page, one must complete the Web Content Manager and Editor Training. This training is supposed to be available before the end of September, 2012. Should it not become available by the end of October, 2012, the project owner shall investigate the possibility of onsite training during a planned visit in late-October – early-November, 2012.

5.3 Project Constraints

Remote execution- Server, software and committee are located at UNCW in Wilmington, NC. Project Owner will be performing the majority of project tasks remotely from Salt Lake City, UT.

Multiple Departments- Committee members span three different departments within UNCW: Geography & Geology; Computer Science; Information Systems & Operations Management.

Technical Support- Project requires cooperation and support from UNCW ITSD.

6.0 PROJECT STRUCTURE APPROACH

Project Owner will conduct bi-weekly meetings with the Project Committee. The project Owner will be responsible for executing and tracking all work agreed upon by the Project Committee.

The Project Committee shall communicate via email, shared documents and telephone and teleconference as necessary.

7.0 PROJECT TEAM ORGANIZATION PLANS

Project Team Role	Project Team Member(s)	Responsibilities
Project Owner	Aaron Caldwell	Oversees project, project execution
Project Committee Member, Client	Dr. Joanne Halls	Determine project requirements, oversee project execution and milestones
Project Committee Leader	Dr. Douglas Kline	Oversee project execution and milestones
Project Committee Member	Dr. Ron Vetter	Oversee project execution and milestones
Project Committee	Dr. Bryan Reinicke	Oversee project execution and milestones

Member		

**APPENDIX B: PROJECT CHARTER AS REVISED TO REFLECT PROJECT
CHANGES AND UPDATES AGREED TO BY PROJECT OWNER AND CLIENT
AS OF OCTOBER, 2013**

Project Name: An ArcGIS Server Application Migration, Application Template Construction and Server Analysis & Design for the University of North Carolina Wilmington Department of Geography & Geology

Department: The Information Systems & Operations Management Department and the Computer Science Department at the University of North Carolina Wilmington

Focus Area: Geographic Information Systems

Product/Process: Server-Based Deployment of ArcGIS Server 10.1 and migration/creation of agreed upon Services and Applications

Prepared By:

Document Owner(s)	Project/Organization Role
Aaron Caldwell	Project Owner
Dr. Joanne Halls	Project Committee Member, Client
Dr. Douglas Kline	Project Committee Leader
Dr. Ron Vetter	Project Committee Member
Dr. Bryan Reinicke	Project Committee Member

Project Charter Version Control:

Version	Date	Author	Change Description
A	July 24, 2011	Aaron Caldwell	Document Created
B	August 1, 2011	Aaron Caldwell	Project Charter Sections Added
C	August 7, 2011	Aaron Caldwell	Project Charter Sections Populated
D	October 2, 2011	Aaron Caldwell	Complete revision for compliance with researched guidelines and input from Dr. Douglas Kline
E	September 16, 2012	Aaron Caldwell	Updated to better reflect client's requirements
F	October 26, 2013	Aaron Caldwell	Updated to reflect changes in project implementation and deliverables

Table of Contents

1. PROJECT PURPOSE
2. PROJECT EXECUTIVE SUMMARY
3. PROJECT OVERVIEW
4. PROJECT SCOPE
 - 4.1 Goals and Objectives
 - 4.2 Departmental Statements of Work (SOW)
 - 4.3 Key Stakeholders

4.4 Project Deliverables

4.5 Deliverables Out of Scope

4.6 Project Estimated Costs & Duration

5. PROJECT CONDITIONS

5.1 Project Assumptions

5.2 Project Issues

5.3 Project Risks

5.4 Project Constraints

6. PROJECT STRUCTURE APPROACH

7. PROJECT TEAM ORGANIZATIONS PLANS

1. Project Purpose

The Project Owner, with the support of the Project Committee, will perform the following services for UNCW Department of Geography and Geology: rebuild of the web mapping applications: Oculina and Riverrun from the previous ArcGIS 9.2 map server to the new ArcGIS 10.1 map server; a sample application template for future GIS application creation; an update to the current web portal to reflect migrated map applications; an analysis & design of the current map server.

2. PROJECT EXECUTIVE SUMMARY

Project Goals: Analyze current setup of ArcGIS Server, create application template, update GIS Web Portal and migrate map applications to new server for UNCW Department of Geography and Geology

Objectives: To better utilize UNCW's ESRI software site license and provide a better platform for GIS instruction.

Scope: Analyze system and application-level requirements to determine a convenient means for deploying services and applications using ArcGIS Server 10.1 in a high quality format that meets project requirements within the agreed upon timeframe.

Assumptions: The required infrastructure has been allocated including space on the server, ESRI ArcGIS Server software and ITSD resources for future maintenance/upkeep.

Risks: Server latency due to excessive traffic, dependency on ITSD for allocation of resources, many tasks being performed at a distance.

Costs: This project is being conducted by the project owner in partial fulfillment of the requirement for M.S. in Computer Science and Information Systems at UNCW. All software resources have already been allocated (see Assumptions). As such, there will be no inherent cost for this project apart from time commitments by project committee members, associated departments and the UNCW Information Technology and Services Division (ITSD).

Timeline: All deliverables agreed upon by the project committee shall be met by December, 2013.

Approach: Project Owner shall conduct meetings with individual Project Committee members as needed. The project Owner will be responsible for executing and tracking all work agreed upon by the Project Committee. The Project Committee shall communicate via email, telephone and in-person meetings as necessary.

Organization:

- Aaron Caldwell (Project Owner)
- Dr. Joanne Halls (Project Committee Member, Client)
- Dr. Douglas Kline (Project Committee Leader)
- Dr. Ron Vetter (Project Committee Member)
- Dr. Bryan Reinicke (Project Committee Member)

3. PROJECT RATIONALE

The UNCW Department of Geography and Geology is currently licensed a software package from ESRI providing several ArcGIS products for GIS instruction. The software package site license includes software for individual workstations (ArcGIS Desktop) as well as a server component (ArcGIS Server) for deployment of Internet-accessible customized map services. Currently the department utilizes the version 10.1 workstation component of this new software package alongside versions 9.2 and 10.0 server components. Migrating previously-built, critical map applications from the old server to the new server, creating an application template and performing an analysis of the new server system, will allow the UNCW Geography Department to better utilize the ESRI Software site license and provide a more consistent platform for GIS course instruction.

4. PROJECT SCOPE

4.1 Goals and Objectives

Goal(s)	Objectives
Migrate Oculina and Riverrun applications from old server to new server	Determine failure mode for map applications inability to run on new server Implement appropriate code modifications for operation on new server Publish updated, functioning map applications on new server and portal
Build sample application for GIS instruction	Use lessons learned in application creation to document process for service/application creation Create template for student application creation
Update ArcGIS Server Web Portal	Modify/update web portal to reflect migrated map applications
Install ArcGIS Server 10.1 and perform an Analysis	Analyze: <ul style="list-style-type: none"> GIS Servers- hosts GIS resources and exposes them to client

and Design of the current system	<p>applications.</p> <ul style="list-style-type: none"> • Web Servers- host's web applications and services using resources on the server. • Data Servers- resources published on server.
----------------------------------	---

4.2 Departmental Statements of Work (SOW)

Departmental SOW	Owner/Prime	Due Date/Sequence
Research and Analysis	Project Owner	July 01, 2011 – September 15, 2012
Configuration and Implementation	Project Owner	September 10, 2012 – November 15, 2013
Advising Resource	Project Committee	August 01, 2011 – December 2, 2012
Project Defense	Project Owner	December 2, 2012

4.3 Key Stakeholders

Stakeholder	Impact to and Participation of Organization
University of North Carolina Wilmington (UNCW)	Host resources physical resources and site license for ArcGIS Server
UNCW Information Technology and Security Division (ITSD)	Site contact for server resources, configuration questions and deployment. ITSD will act as a technical resource for the ArcGIS Server implementation following the project's completion.
UNCW Department of Computer Science	Faculty and staff will be consulted for server implementation expertise.
UNCW Department of Information Systems and Operations Management	Faculty and staff will be consulted for server implementation expertise.

UNCW Department of Geography and Geology	Faculty and Staff will be the primary administrators and users of ArcGIS Server Software. Department members will be consulted for input regarding initial configuration.
--	---

4.4 Project Deliverables

Milestone	Deliverable
Web application migration	Oculina and Riverrun confirmed to function on the new server with a similar level of functionality as permitted in the new environment
Sample application template	A sample web application from Esri's provided JavaScript templates will be modified to allow "novice developer" code modification by UNCW GIS students for application creation. This will be accompanied by instructions for proper application creation.
Web portal update	<p>Completion of the GIS portal update will be met by completion of the following two requirements:</p> <ol style="list-style-type: none"> 4. (pre-requisite to website modification) Web Content Manager and Editor Training completed. This training is mandated by UNCW for staff, faculty and students who wish to modify websites representing either the university or one of its constituent colleges 5. Successful application migration and subsequent posting of publicly accessible URLs with accompanying descriptions. 6. Link for sample template
Installation of ArcGIS 10.1, Analysis & Design of System	<p>ArcGIS 10.1 shall be installed to perform at the same level of functionality as that of the 10.0 server system.</p> <p>For the analysis & design, the following items shall be delivered for a complete conceptual view of the web server:</p> <ul style="list-style-type: none"> • Actor Diagrams • Instructions for use (Use cases) • Business process models for service and application creation on each

	of the three servers <ul style="list-style-type: none"> System component analysis, wireframes for application design
--	---

4.5 Project Estimated Duration

Milestone	Date Estimate	Deliverable	Confidence Level
Project Definition	September 28, 2012	Define project goals Determine configuration options Estimate required resources	Medium
Analysis & Design of Web Server	February 20, 2013	Actor Diagrams Business process models for service and application creation on each of the three servers System component analysis, wireframes for application design	Medium
Web application migration	November 10, 2013	Oculina and Riverrun web applications confirmed to function on new server	Medium
Sample application creation	November 11, 2013	Sample application template for student use created and posted to GIS Portal	High
Web portal update	November 15, 2013	Links/descriptions of web applications posted to UNCW's GIS portal	Medium
Project completion	December 02, 2013	Determine that project meets stakeholder requirements within given constraints Project defense	Medium

4.6 Deliverables Out of Scope

- Other data or documentation linked or made accessible via web portal.
 - Instructions for administration of the server and applications
 - Migration of services and applications other than Riverrun and Oculina
-

5. PROJECT CONDITIONS

5.1 Project Assumptions

- UNCW ITSD is currently in possession of an ESRI site license with rights for operation of ArcGIS Server Enterprise Edition as previously confirmed.
- UNCW Department of Geography and Geology and UNCW ITSD will handle user authentication for web portal and ArcGIS Server use.
- UNCW ITSD will provide ArcGIS Server support following project completion.

5.2 Project Risks

#	Risk Area	Likelihood	Risk Owner	Project Impact-Mitigation Plan
1	Server performance latency	Low	Aaron Caldwell/ ITSD	Analysis and Design could reveal disadvantages for single machine deployment which could impact application development. Project owner will work with ITSD to increase system resources should this situation arise.
2	ITSD Assistance	High	Aaron Caldwell/ ITSD	Remote implementation may slow interactions however the project committee member have the physical presence for face-to-face communication as well as the leverage within the UNCW system to assist with ITSD as necessary.

3	Performing Project Research and Work at a Distance	High	Aaron Caldwell	Any research or operations that can't be performed remotely will be accomplished through a combination of the following in descending order of frequency: <ul style="list-style-type: none"> • Periodic visits to the UNCW campus • Assistance from ITSD • Assistance from Project Committee

5.3 Project Constraints

Remote execution: Server, software and committee are located at UNCW in Wilmington, NC. Project Owner will be performing the majority of project tasks remotely from multiple locations including Milwaukee, WI, Salt Lake City, UT and Richmond, VA.

Multiple Departments: Committee members span three different departments within UNCW: Geography & Geology; Computer Science; Information Systems & Operations Management.

Technical Support- Project requires cooperation and support from UNCW ITSD.

6.0 PROJECT STRUCTURE APPROACH

Project Owner will conduct meetings with Project Committee members as needed. The project Owner will be responsible for executing and tracking all work agreed upon by the Project Committee. The Project Committee shall communicate via email, shared documents and telephone and teleconference as necessary.

7.0 PROJECT TEAM ORGANIZATION PLANS

Project Team Role	Project Team Member(s)	Responsibilities
Project Owner	Aaron Caldwell	Oversees project, project execution
Project Committee Member, Client	Dr. Joanne Halls	Determine project requirements, oversee project execution and milestones
Project Committee Leader	Dr. Douglas Kline	Oversee project execution and milestones
Project Committee Member	Dr. Ron Vetter	Oversee project execution and milestones
Project Committee Member	Dr. Bryan Reinicke	Oversee project execution and milestones

APPENDIX C: REQUIREMENTS THAT CAME OUT OF PROJECT PROPOSAL

The following is a list of requirements identified during the project proposal that shall be completed in addition to the original requirements outlined for this project:

Reference #	Description	Priority	Current Status	Actions Performed
PR-1	Documentation/ description of the current server setup	Medium	Complete November, 2012	Documented in pp. 17- 20.
PR-2	A list of the applications (from Dr. Halls) that need to be converted, with a description of each; separated into two groups: critical and optional.	High	Originally completed in November, 2012. Updated requirements to describe testing methods (Kline) completed March, 2013	It was determined by Dr. Halls that only 2 critical map services are required for migration: Oculina and Riverrun (greater detail on p. 15). Expected functionality of applications listed in Appendix C. Testing forms can be found in Appendix D. Completed test forms will be added following migration.
PR-3	Critical aspects of appearance and functionality that must be present in migrated services when activated on new server	Medium	Completed March, 2013.	See Appendix E for wireframes.
PR-4	1 application converted, i.e., pilot the conversion process; a detailed log	Low	Final application functionally complete by October, 2013	Oculina converted first, basic framework completed in July, 2013

	of the conversion steps: every command, every option changed, every code block before and after, etc.				
	PR-5	A log of how long it took to do the conversion	Low	Completed November, 2013	Log kept for application conversion, published in Appendix.
	PR-6	A Use Case for the “sample application” you plan to do in the final project	Medium	Completed March, 2013	Created Actor Diagrams, instructions for students and business process models, one for Sample Service and another for Migrated Services in Appendix G. They’re functionally very similar as they fulfill similar roles for users
	PR-7	Sequence Diagrams for cookbook/walkthroughs	Medium	Completed March, 2013	BPMN Diagram in Appendix F
	PR-8	Actor Diagram, for the current server setup, the center system components being the user interfaces that stakeholders interact with	Medium	Completed March, 2013	Server-view actor diagram in Appendix G

PR-9	System Diagram, to describe the current server setup, and if they are different, for the converted web apps versus the new web apps	Medium	Completed November, 2012	A server system diagram is included on p. 18 of this document. Both the converted web apps and the sample web app will require the same server setup as described on p. 18. Modify slightly for new server setup.
PR-10	List of critical requirements for completion of sample Map Service	Medium	Completed March, 2013	A list of critical requirements is briefly referenced on p. 27, actual list may be found in Appendix H.
PR-11	Rationale for analysis & design methodology	Medium	Completed March, 2013	Dr. Vetter requested a brief rationale for my choice of OOA. That rationale is presented in the 1 st paragraph on p. 19 and shall be explored further in the course of the analysis & design of the system.

APPENDIX D: MAP SERVICE ELEMENTS IDENTIFIED FOR POTENTIAL MIGRATION

The following Map Service elements were identified for potential migration to the new ArcGIS Server:

Map Service/ Application	Critical application elements identified for potential migration
Oculina	<ul style="list-style-type: none"> • Esri ArcGIS Template Features <ul style="list-style-type: none"> ○ Legend indicating topography ○ Zoom in/out ○ Hand grab/drag feature ○ Distance measurement tool ○ Rectangle select ○ Select line/polygon feature ○ Submit query • Map-specific data <ul style="list-style-type: none"> ○ 1 Map, 9 Layers <ul style="list-style-type: none"> ▪ Dive logs ▪ Bottom Habitat ▪ UTM Grid (5 min interval) ▪ Satellite HAPC ▪ Main HAPC ▪ Western Bathymetry ▪ Eastern Bathymetry ▪ Florida • Web-specific data <ul style="list-style-type: none"> ○ Top bar with project title ○ Side bar containing Esri tools

- Esri ArcGIS Template Features
 - Legend indicating topography
 - Compass
 - Scale
 - Inset view of entire layer
 - Zoom in/out
 - Zoom to full-extent
 - Zoom to specified extent
 - Back to previous zoom extent
 - Hand grab/drag feature
 - Distance measurement tool
 - Rectangle select
 - Select line/polygon feature
 - Delete point
 - Print map
- Map-specific data
 - 13 Maps
 - Overview of River Run GIS (11 layers)
 - All stations
 - Swine
 - Cattle
 - Horses
 - Poultry
 - NPDES Permit
 - Roads
 - Rivers
 - Hydrography
 - Municipalities
 - Basin Boundaries

-
- Dissolved Oxygen, Chlorophyll A, Conductivity, Fecal Coliform, Fish species of the lower Cape Fear, Nitrate and Nitrite, Orthophosphate, pH, Total Phosphorus, Salinity, Temperature, Turbidity (Each has 60 layers corresponding to Monthly data (1 layer/month) from June, 1995- May, 2000)
 - Counties
 - Rivers
 - Web-specific data
 - Top bar with project title
 - Side bar containing Esri tools
-

APPENDIX E: WEB APPLICATION FUNCTIONALITY AND ELEMENTS CHECKLIST

Oculina Checklist:







































Element	Present	Not Present	Comment	Initial
Legend indicating topography	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Compass	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC
Scale	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC
Inset view	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Zoom in/out	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Zoom to full extent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	No longer necessary	AC
Zoom to specified extent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	No longer necessary	AC
Return to previous zoom extent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	No longer necessary	AC
Hand grab/drag	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Distance measurement	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Rectangle select features	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Select line/polygon feature	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC



Submit query	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Print map	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC
Dive logs layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Bottom habitat layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Satellite HAPC layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Main HAPC layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Western Bathymetry	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Eastern Bathymetry	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Florida basemap	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Top bar with project title	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Side bar containing Esri tools	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC

Riverrun Checklist:

Element	Present	Not Present	Comment	Initial
Legend indicating topography	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Compass	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC

Scale	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC
Inset view	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Zoom in/out	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Zoom to full extent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	No longer necessary	AC
Zoom to specified extent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	No longer necessary	AC
Return to previous zoom extent	<input type="checkbox"/>	<input checked="" type="checkbox"/>	No longer necessary	AC
Hand grab/drag	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Distance measurement	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Rectangle select features	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Select line/polygon feature	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC
Submit query	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Print map	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Not critical but can easily be added in the future.	AC
All stations layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Swine layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Cattle layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Horses layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC
Poultry layer	<input checked="" type="checkbox"/>	<input type="checkbox"/>		AC

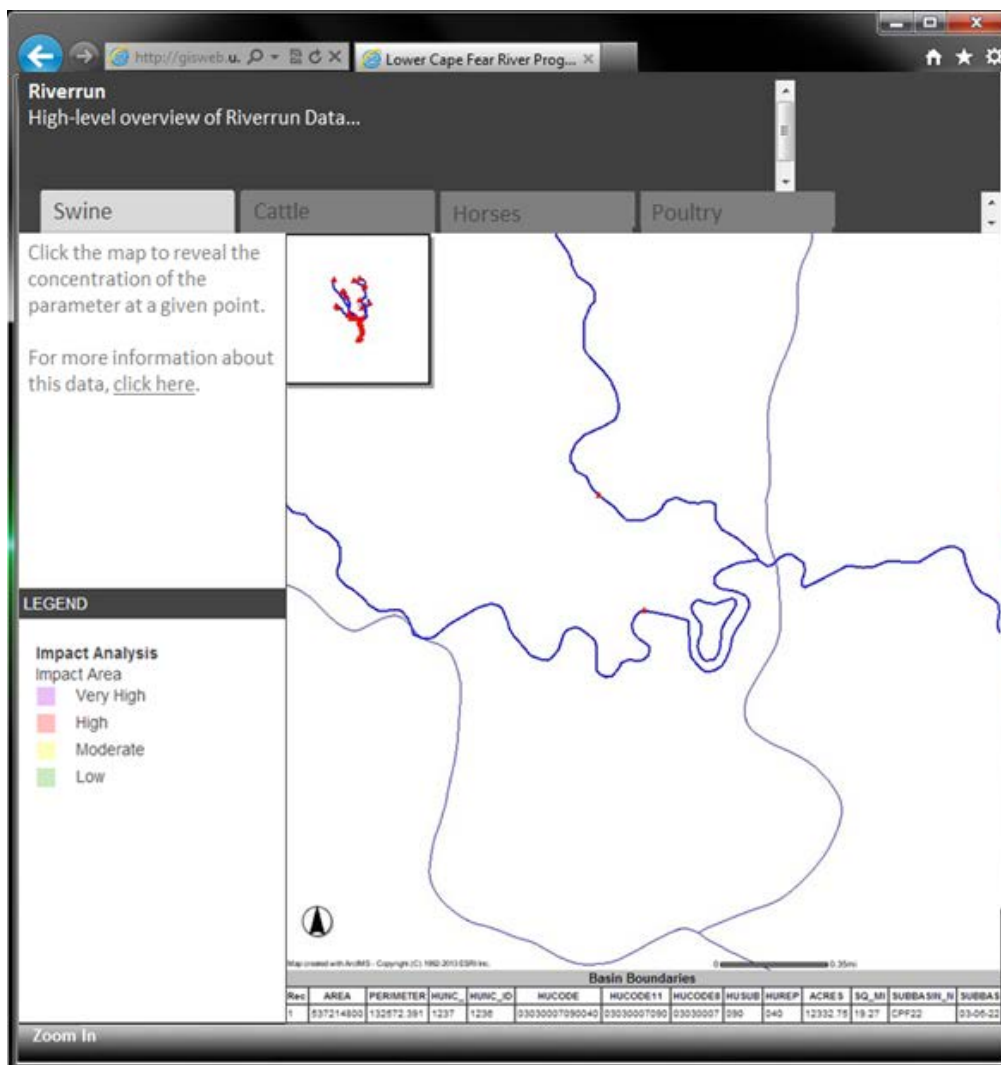
NPDES Permit layer				AC
Roads layer				AC
Rivers layer				AC
Hydrography layer				AC
Municipalities layer				AC
Basin Boundaries layer				AC
Dissolved Oxygen layer				AC
Chlorophyll A layer				AC
Conductivity layer				AC
Fecal Coliform layer				AC
Fish species of the lower Cape Fear layer				AC
Nitrate and Nitrite layer				AC
Orthophosphate layer				AC
pH layer				AC
Total Phosphorus layer				AC
Salinity layer				AC
Temperature layer				AC
Turbidity layer				AC
Top bar with project title				AC

layer				
Side bar containing Esri tools layer				AC

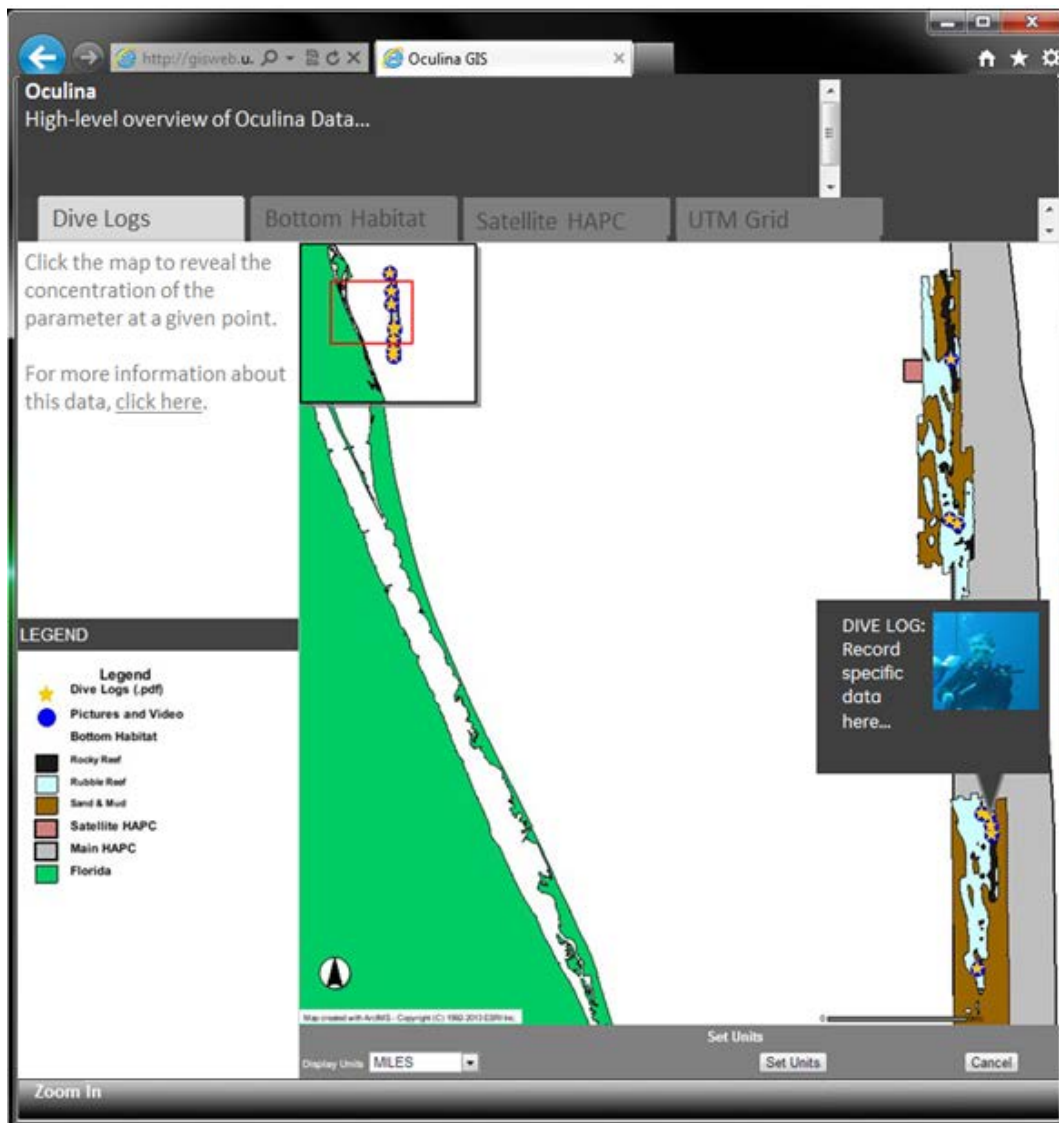
APPENDIX F: WIREFRAMES FOR MIGRATED APPLICATIONS

Both maps are loaded in the wireframes in their current state however a better basemap (background map) may be easily added for the final product. There are several template designs available with some flexibility regarding controls. For consistency both Oculina and Riverrun will be imported to similar formats. There may be some variation in the colors and controls used in the final products to accommodate each project's unique design.

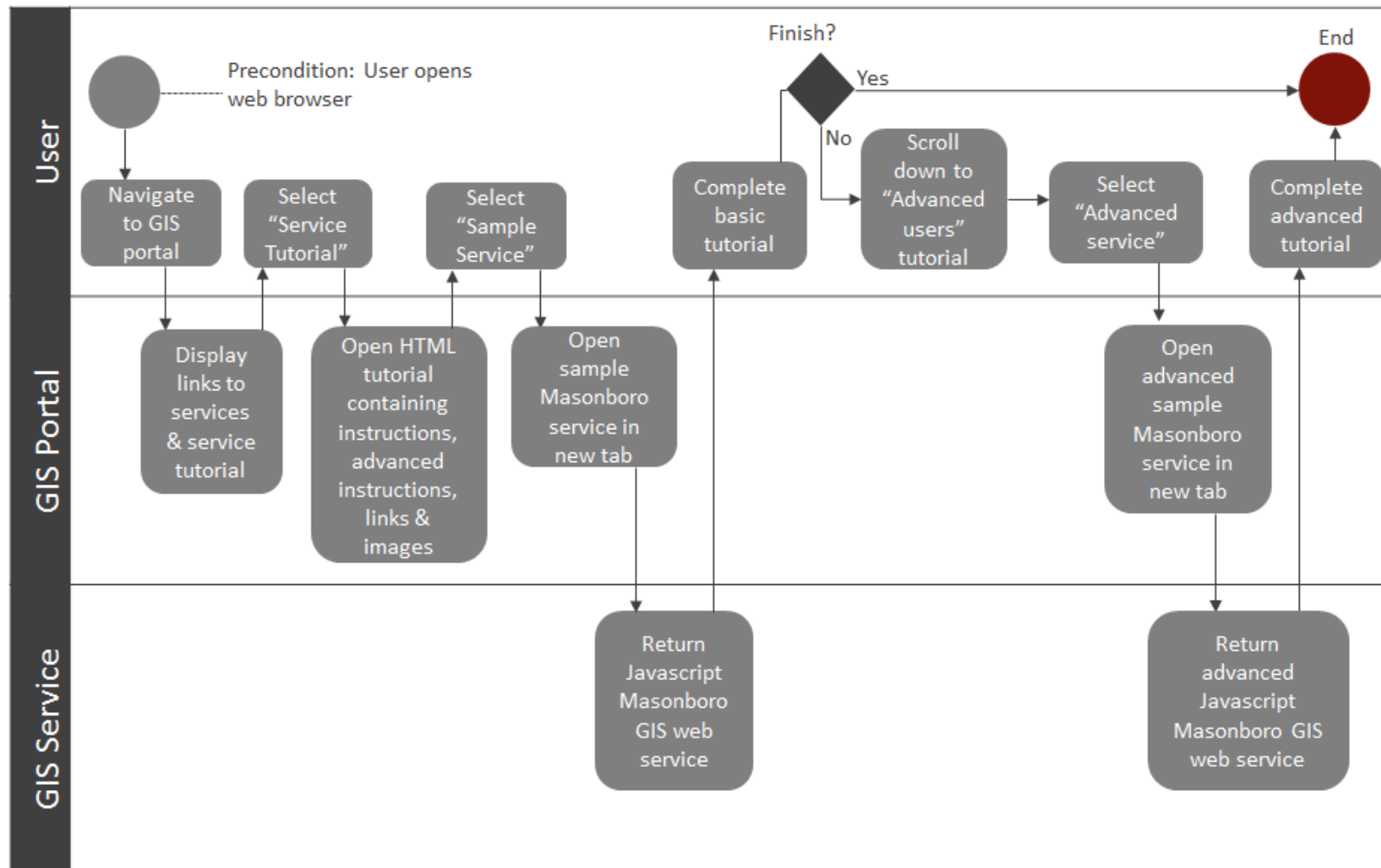
Riverrun:



Oculina:

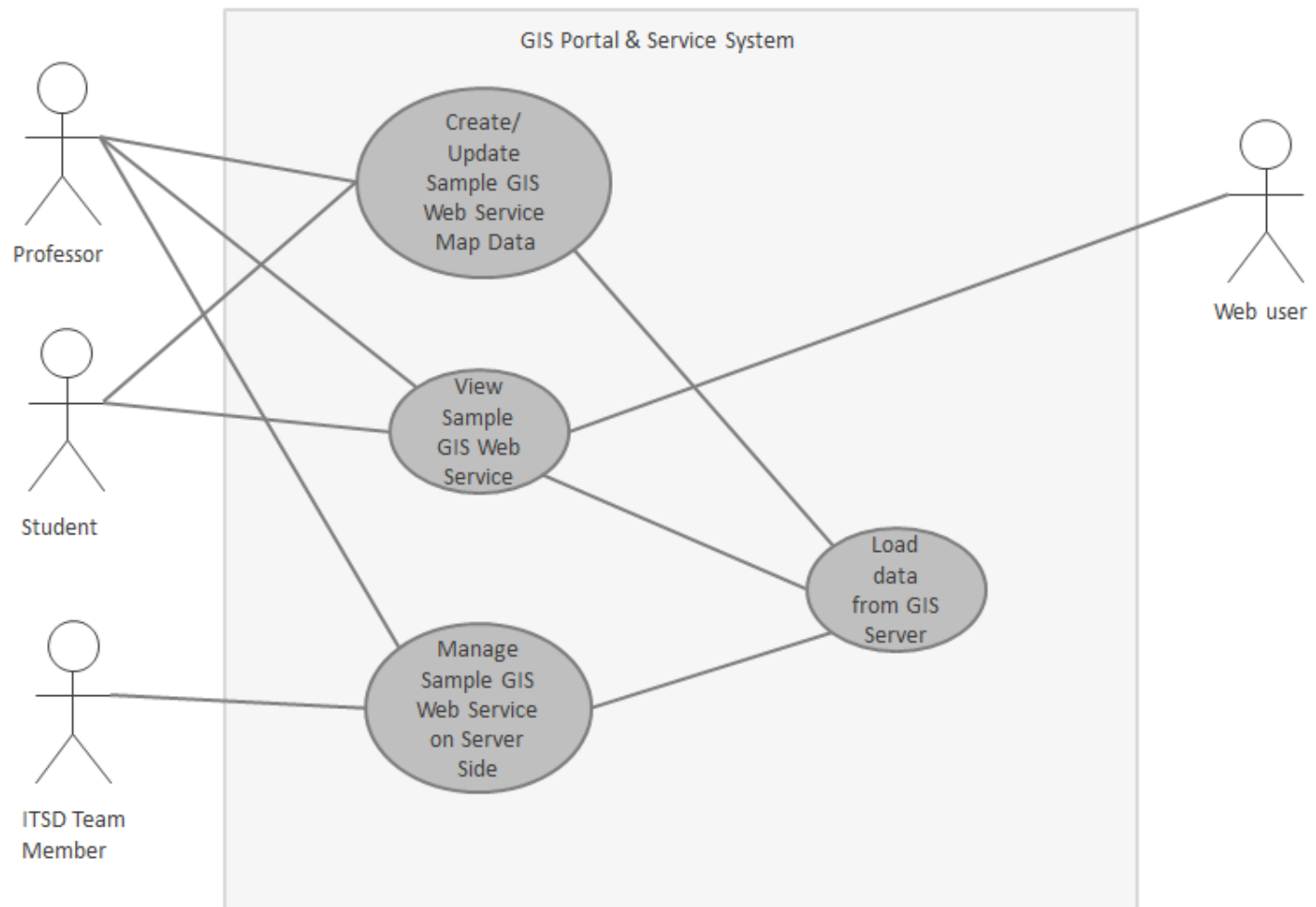


APPENDIX G: ORIGINAL BUSINESS PROCESS MODEL & NOTATION (BPMN) DIAGRAM AS PROPOSED FOR SAMPLE SERVICE (LATER REPLACED BY APPLICATION WORKFLOW IN APPENDIX N)

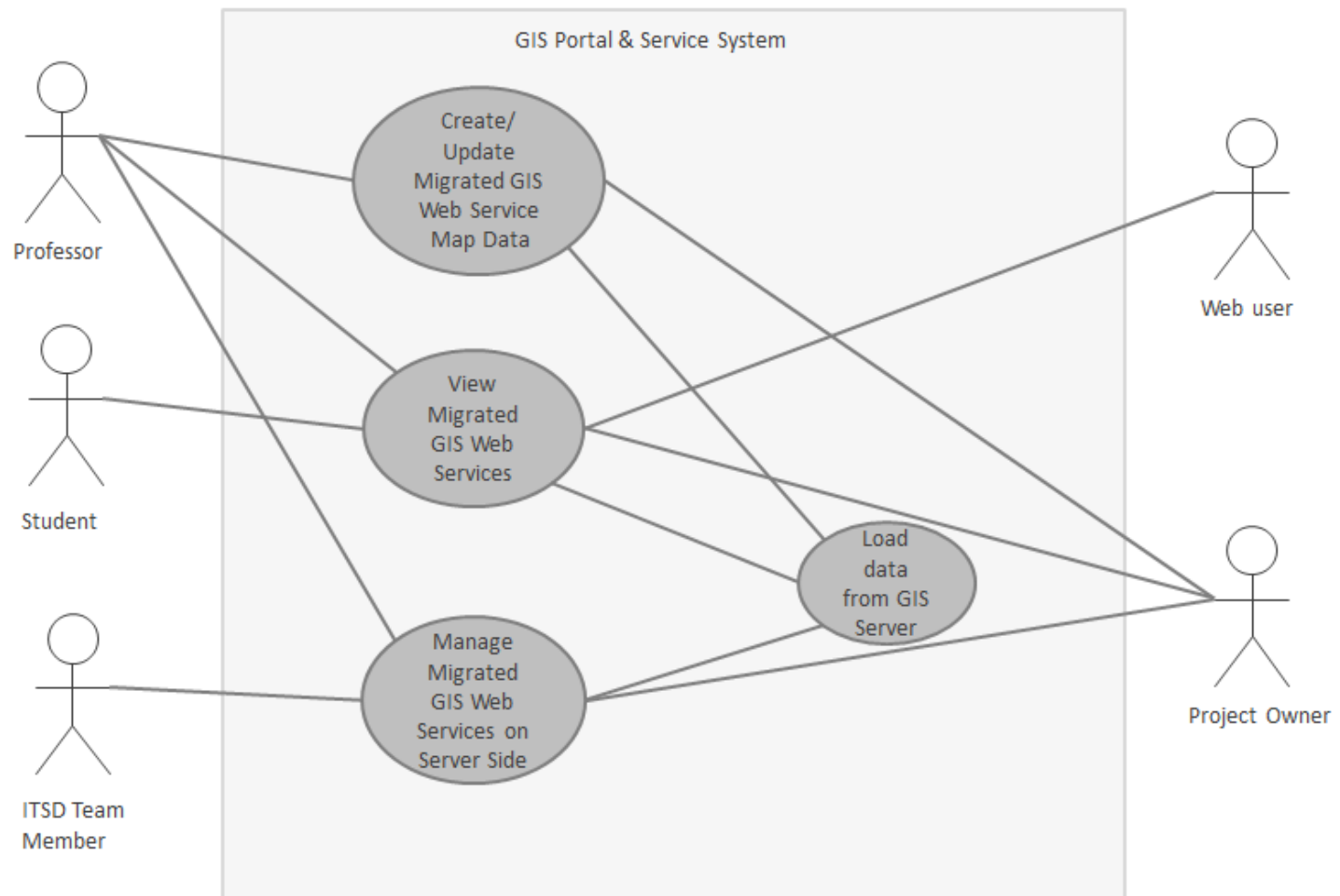


APPENDIX H: ACTOR DIAGRAMS

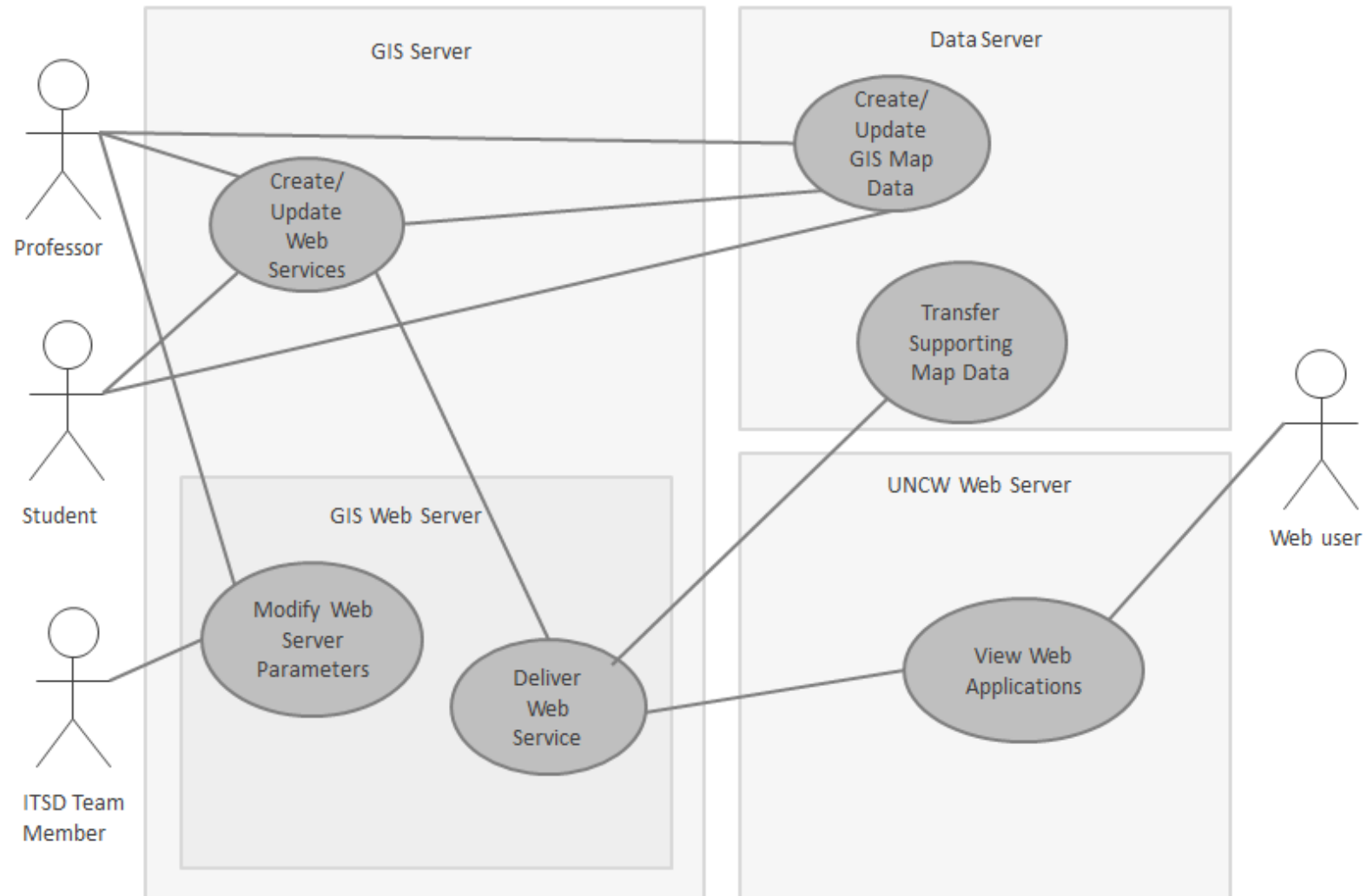
Sample GIS Map Service:



Migrated GIS Services:

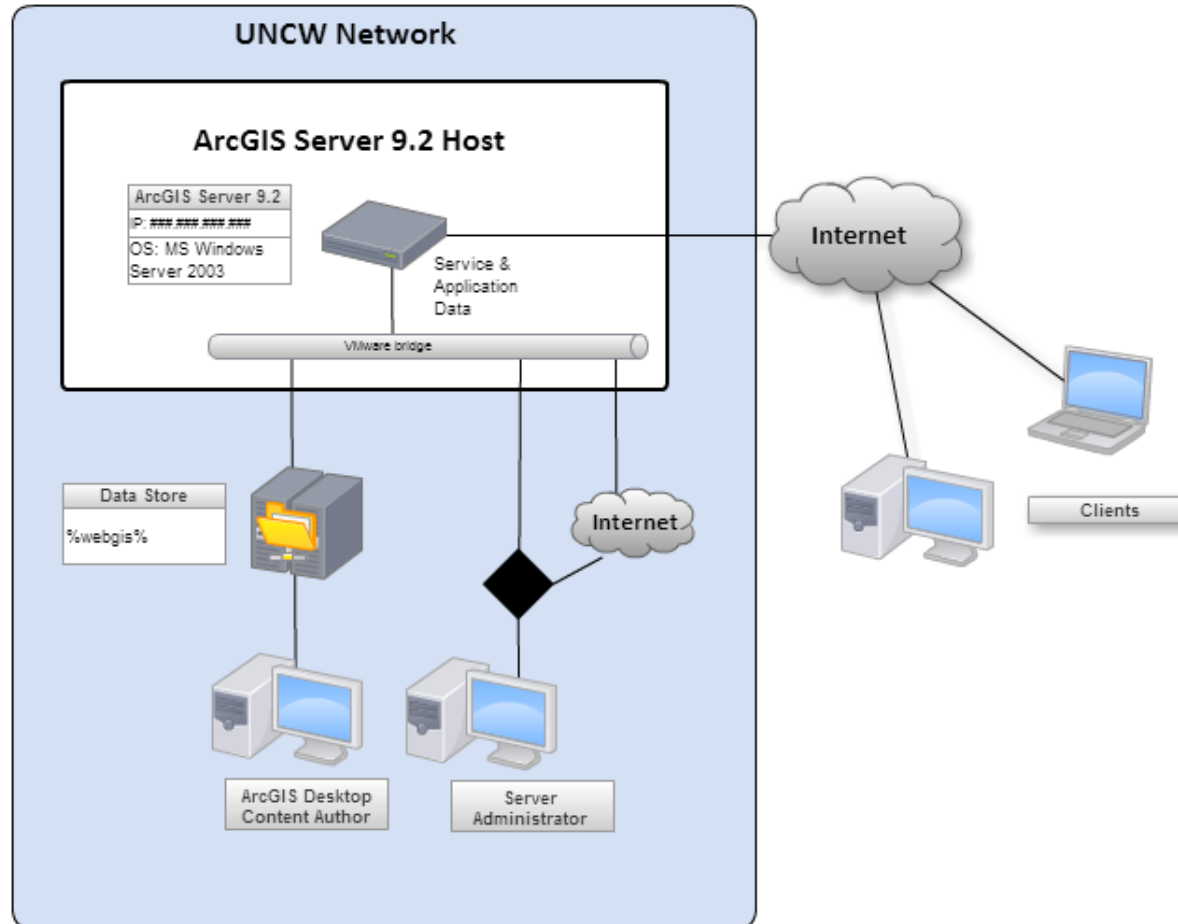


Server-view Services/Applications:

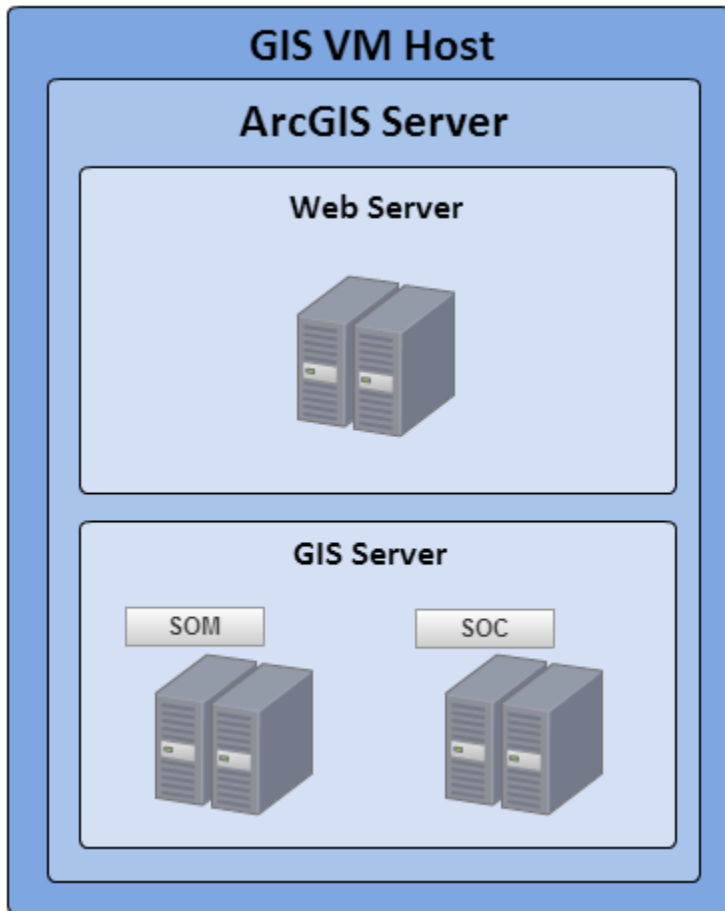


APPENDIX I: ARCGIS 9.2 SERVER SETUP

A. Application and service publishing configuration:

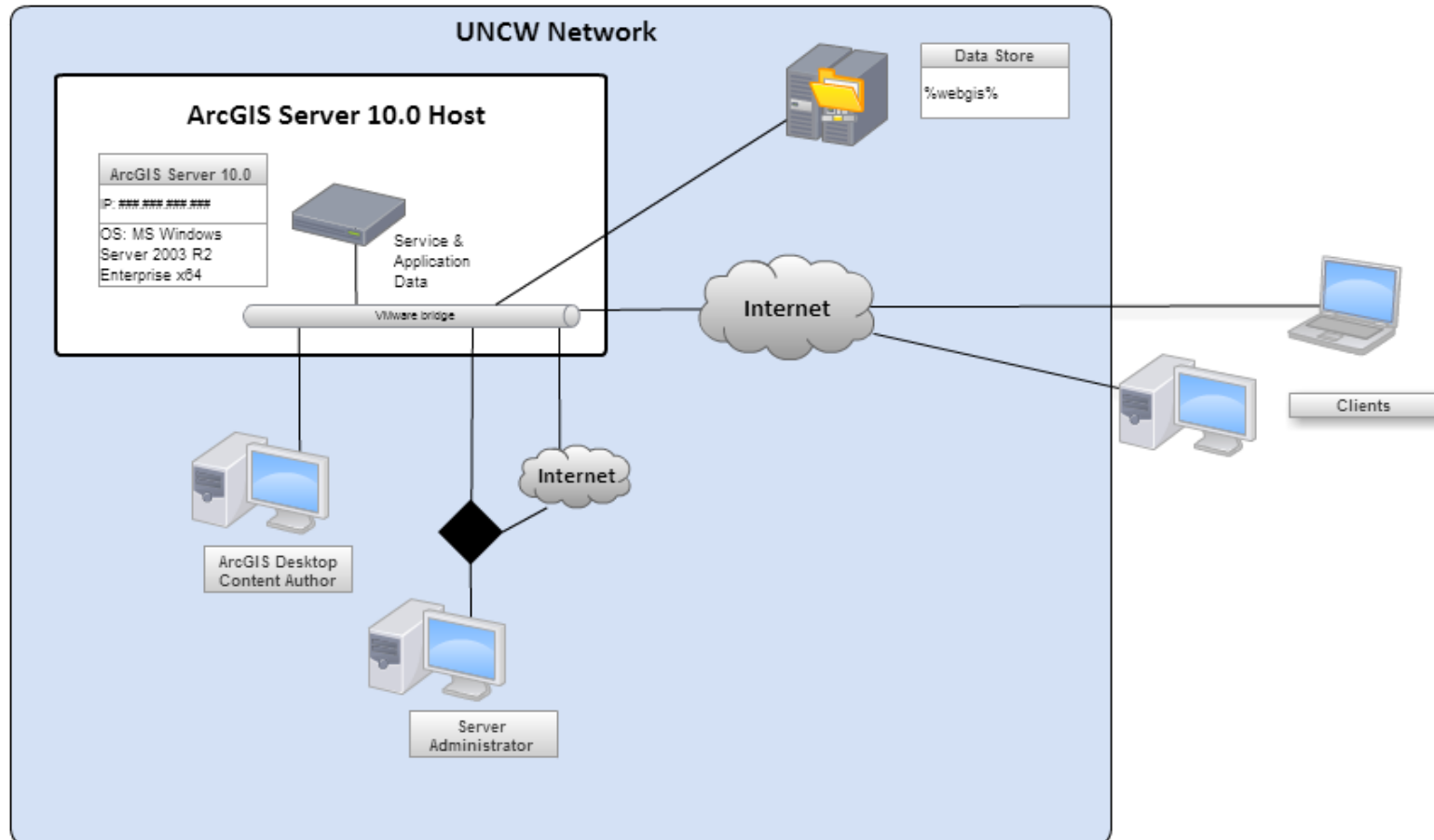


B. GIS VM setup:

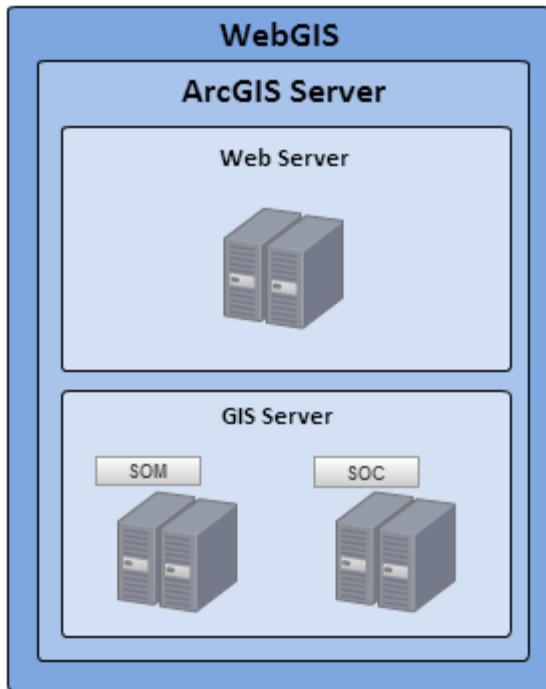


APPENDIX J: ARCGIS 10.0 SERVER SETUP

A. Application and service publishing configuration:

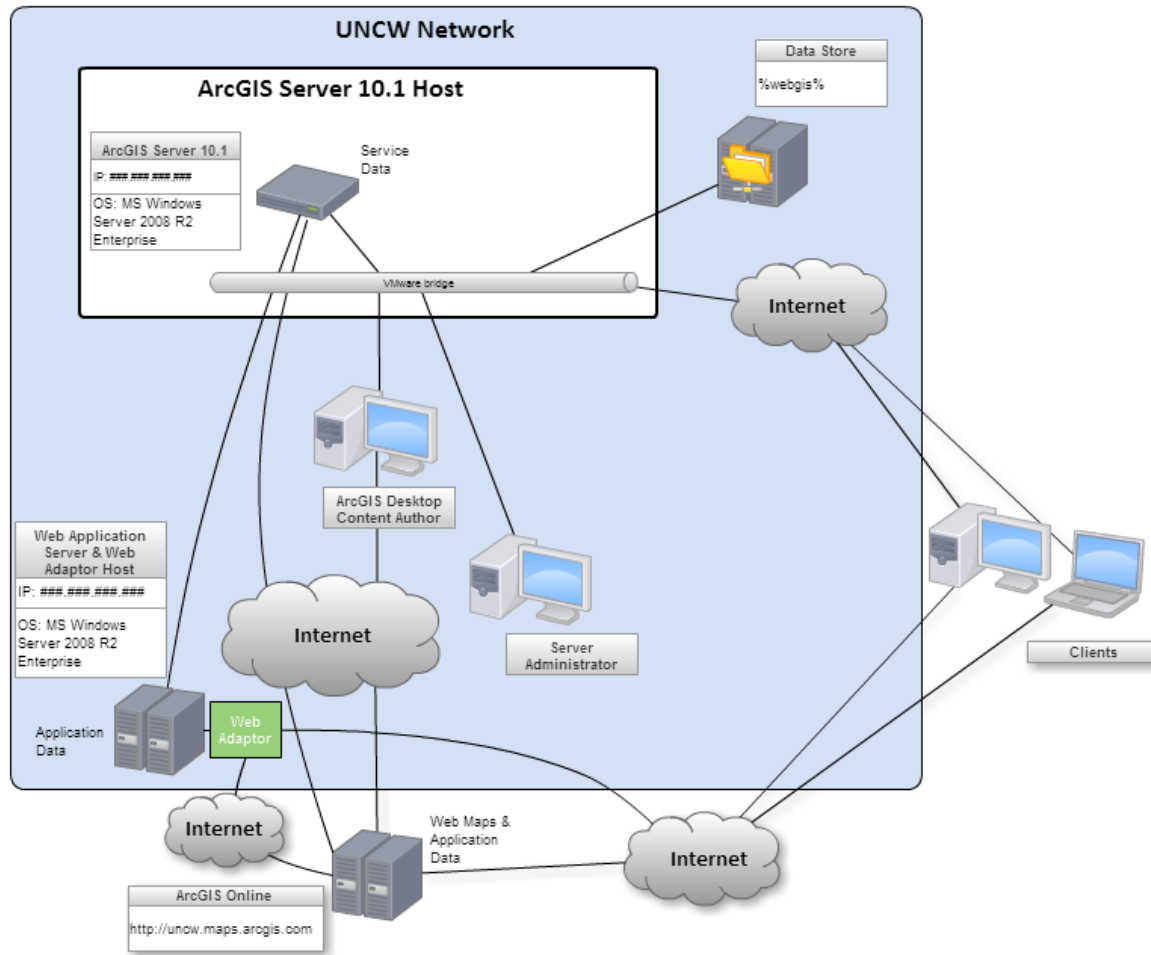


B. GIS VM setup:

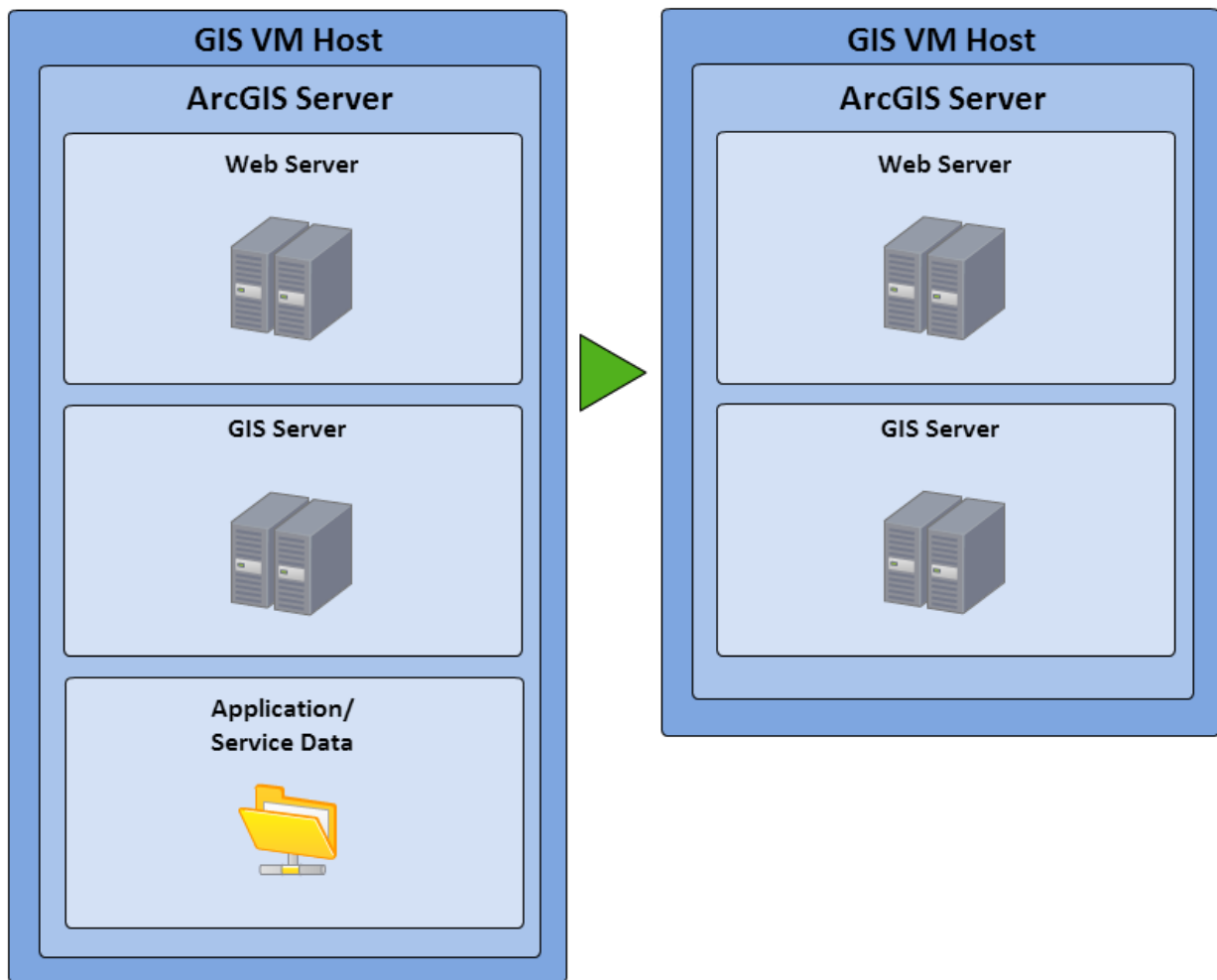


APPENDIX K: ARCGIS 10.1 SERVER SETUP

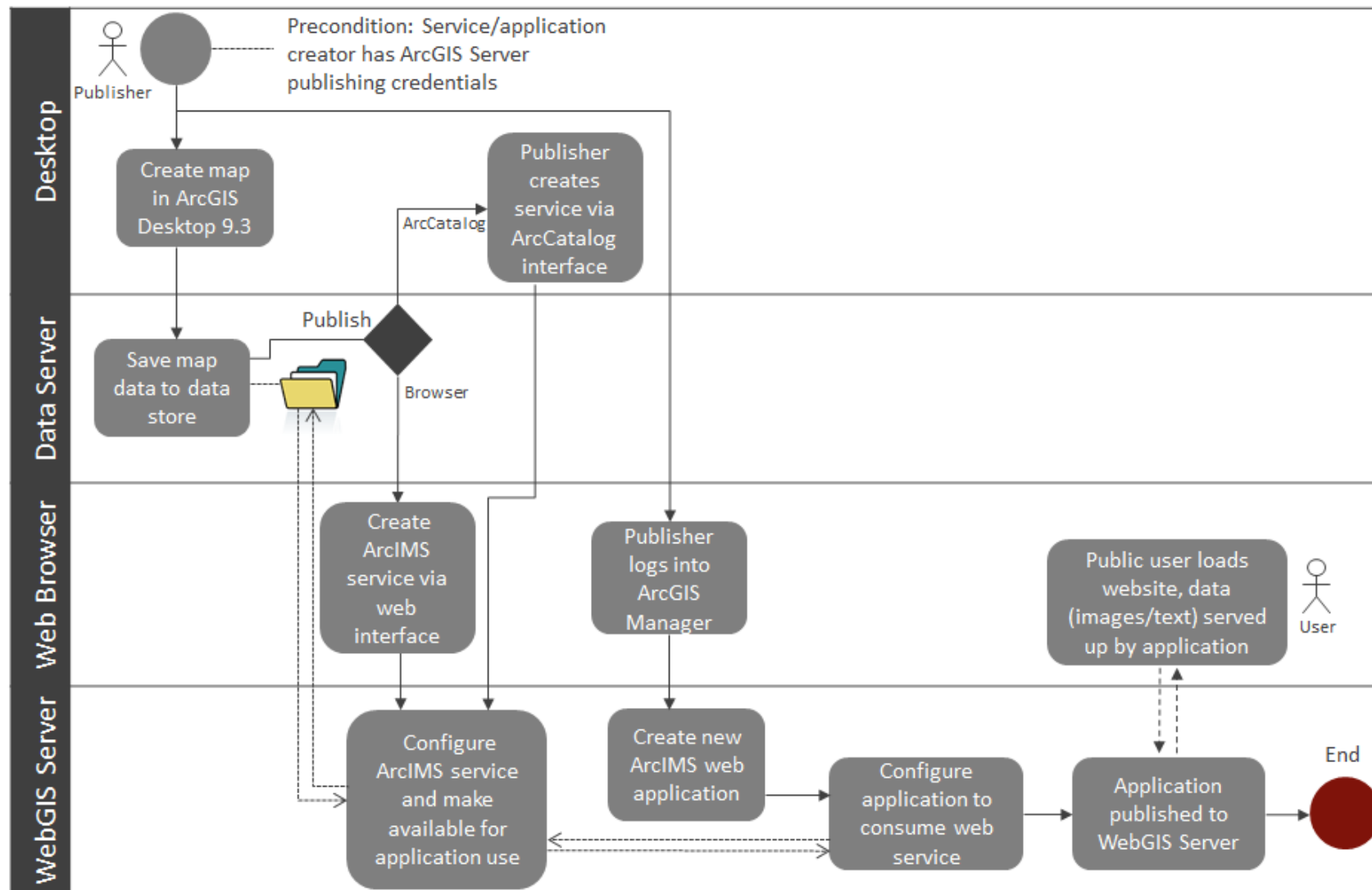
A. Application and service publishing configuration:



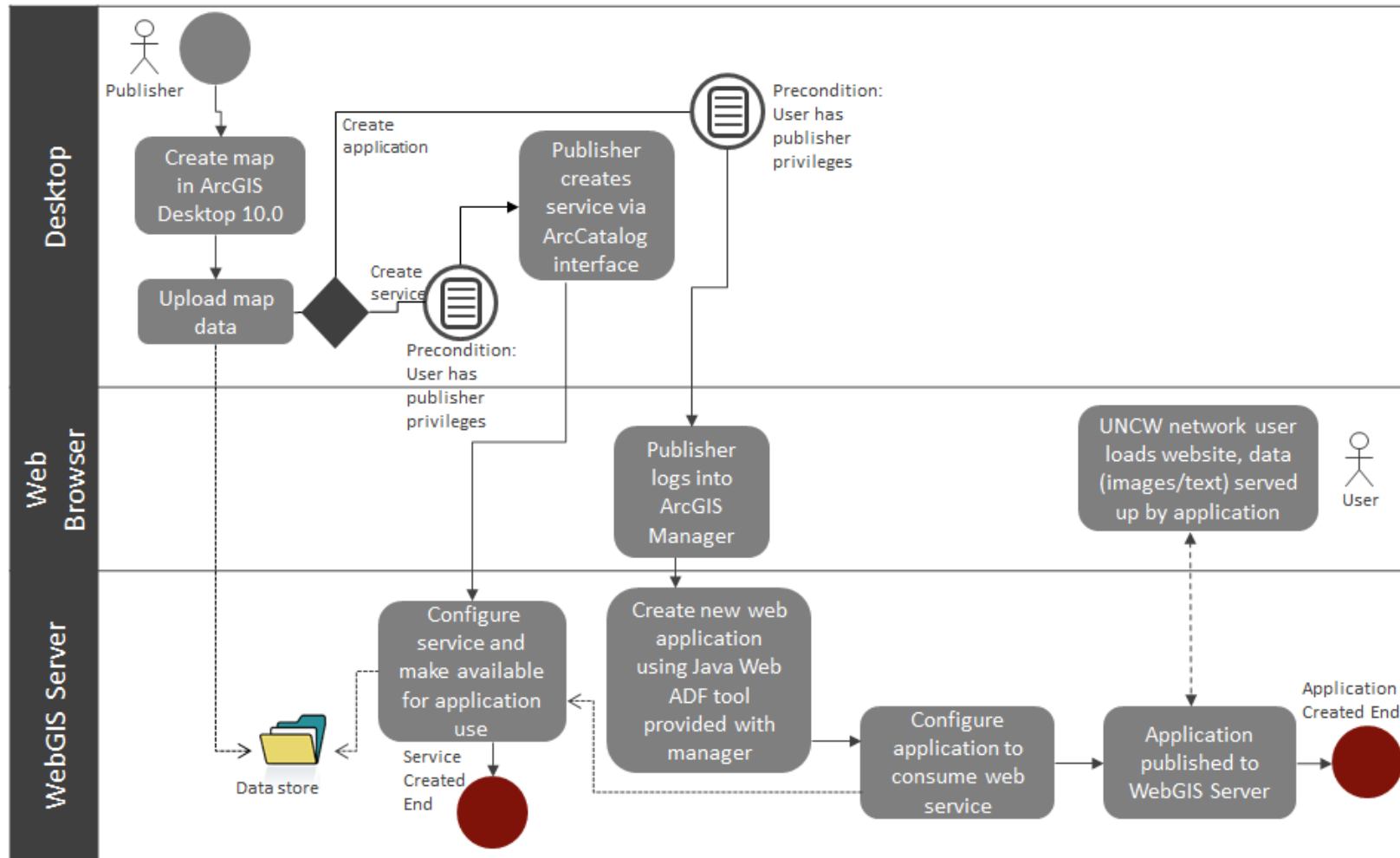
B. GIS VM setup:



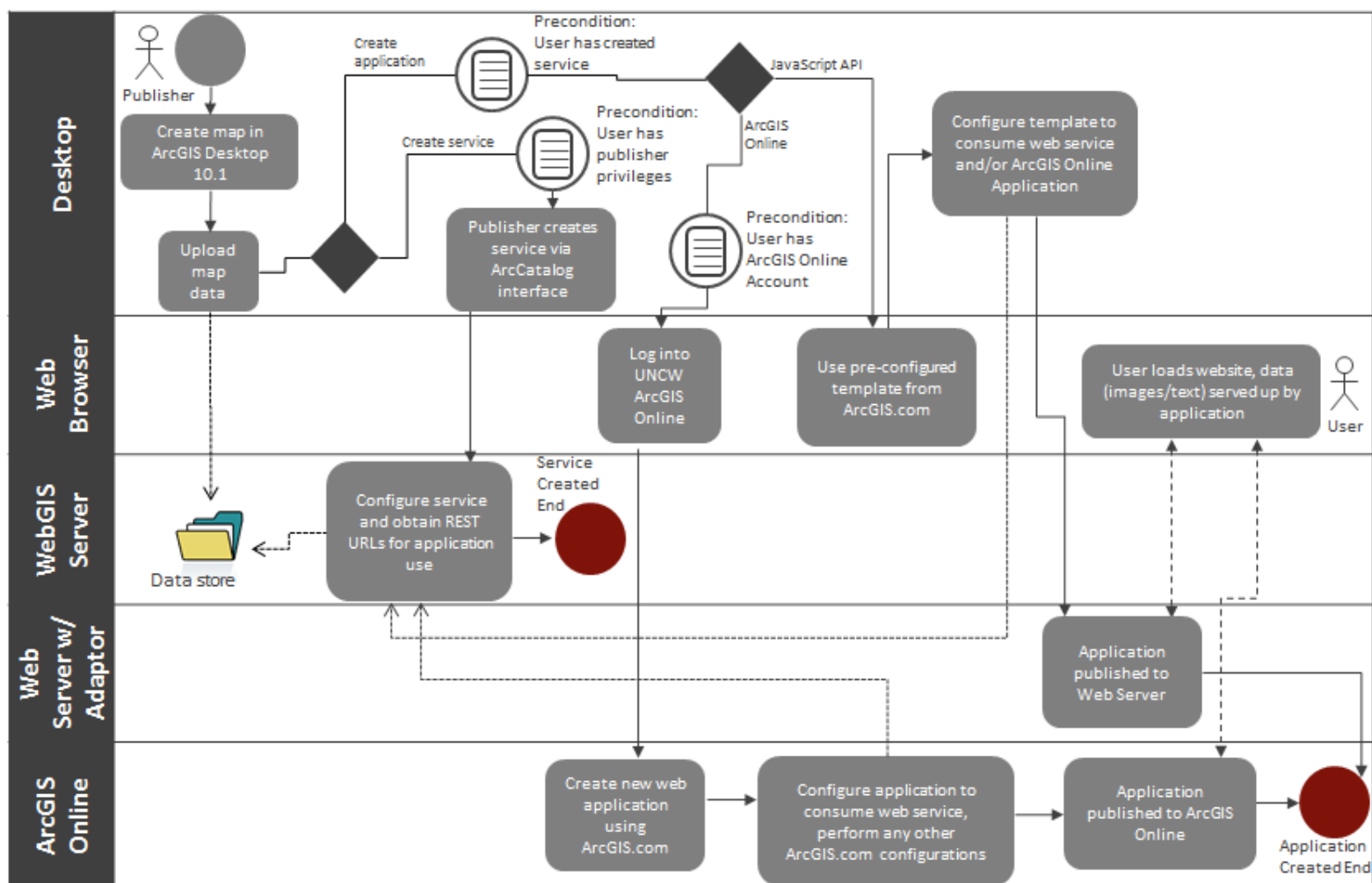
APPENDIX L: ARCGIS 9.2 APPLICATION CREATION WORK FLOW



APPENDIX M: ARCGIS 10.0 APPLICATION CREATION WORK FLOW



APPENDIX N: ARCGIS 10.1 APPLICATION CREATION WORK FLOW



APPENDIX O: RESULTS OF ARCGIS 10.1 SERVER HOST READ/WRITE TESTS TO LOCAL VS. NAS DRIVE

		Local (C:\)								Network Attached Storage (\\arcshare)							
		Seq		512k		4k		4k QD32		Seq		512k		4k		4k QD32	
Date	Time	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write	Read	Write
10/11/2013	12pm	67.2	89.3	29	43	0.32	0.66	1.18	0.91	102	150	84.9	141	9.9	8.81	21.3	19.1
	6pm	71.8	89.1	31.1	51.1	0.4	0.81	4.67	2.47	110	193	109	181	11	10.2	41.1	27.5
10/14/2013	4pm	68.4	89.3	27.5	42.7	0.33	0.73	1.26	1.64	132	187	136	183	13.7	11.9	37.7	28.3
10/15/2013	3:30pm	67.3	85.5	29.5	44.4	0.34	0.71	1.29	1.08	124	188	132	176	13.3	12.4	42.6	27.9
10/16/2013	1:30pm	59.4	91.6	26.1	44.1	0.34	0.68	1.06	0.88	134	198	134	186	13.1	11.7	47.2	27
	4:30pm	69.3	107	27.8	49.6	0.39	0.69	1.43	1.23	136	193	130	178	13	12.1	45	24.8
	Average:	67.2	92	28.5	45.8	0.35	0.71	1.81	1.37	123	185	121	174	12.3	11.2	39.2	25.7

APPENDIX P: MAP SERVICE CREATION INSTRUCTIONS

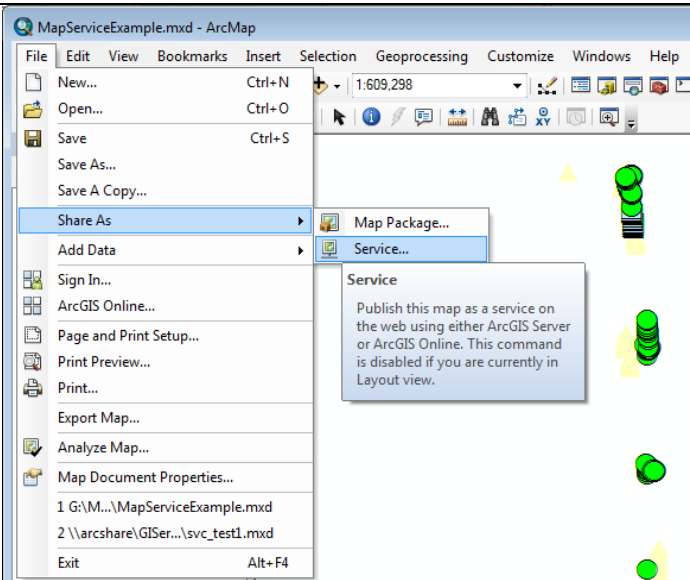
Create a map service

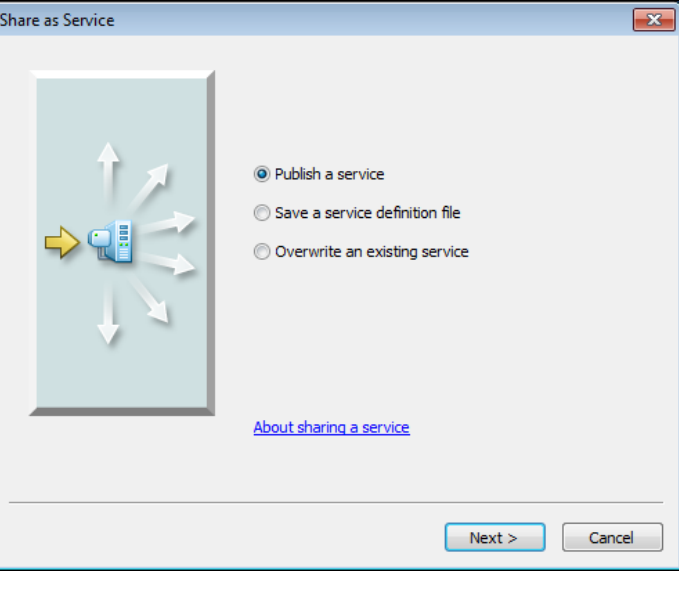

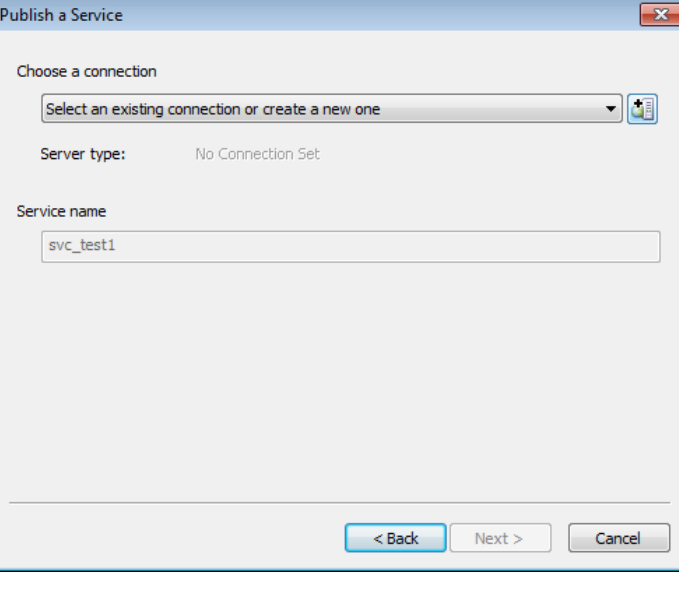
Modified and adapted from Esri's instructions* for use by UNCW Department of Geography & Geology

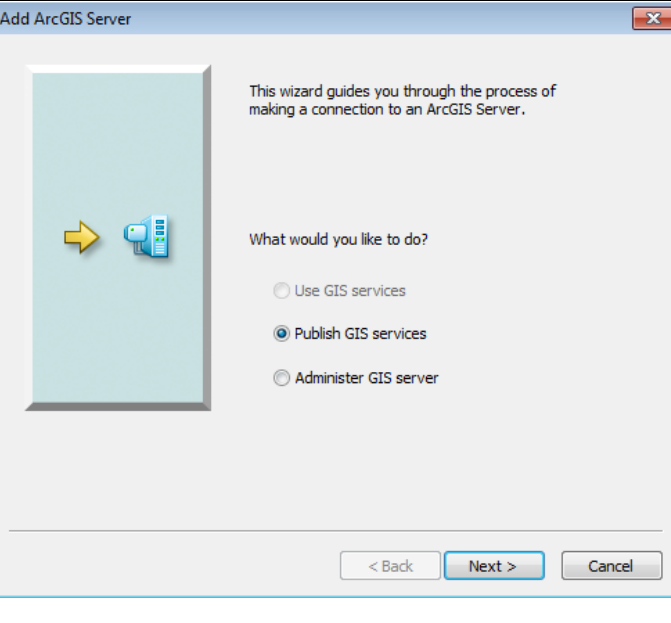
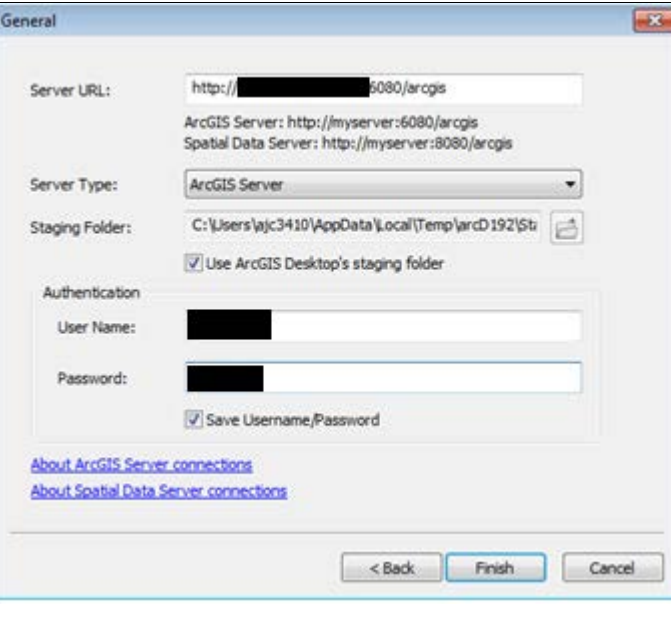
Prerequisite checklist:

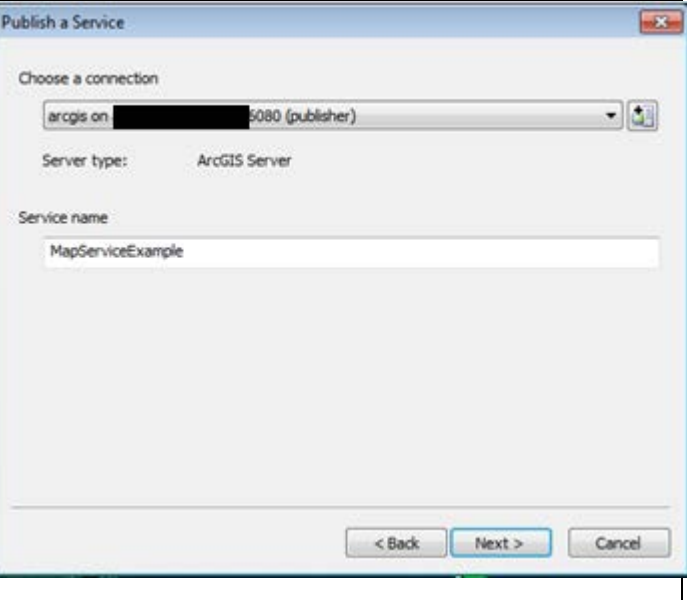
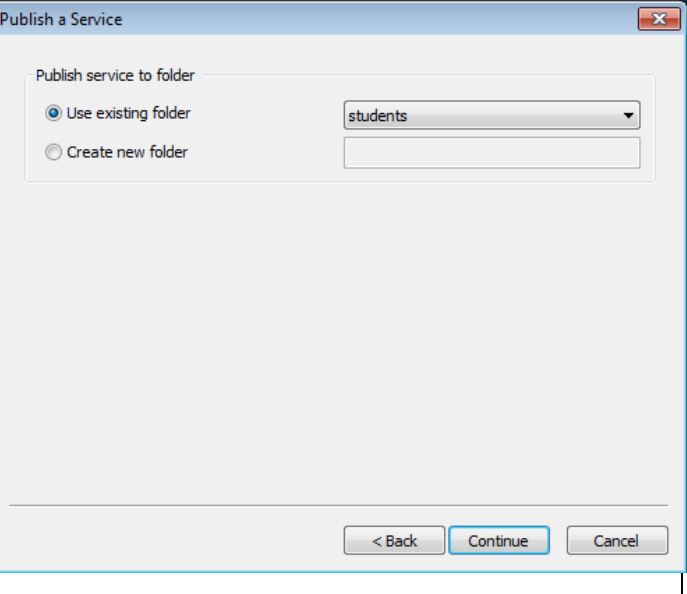
- Have you placed your final map service data in the designated folder on the shared drive? (%webgis%)
 - This includes all service supporting map data: shapefiles, geodatabases, etc. NOT .mxd files
- Have you saved your final .mxd file in your \\arcshare\giserver\users folder?
- Several classes have access to the %webgis% folder, if you want to keep a backup copy of your data on your G drive, you're welcome to make a copy and save it to \\arcshare\giserver\users

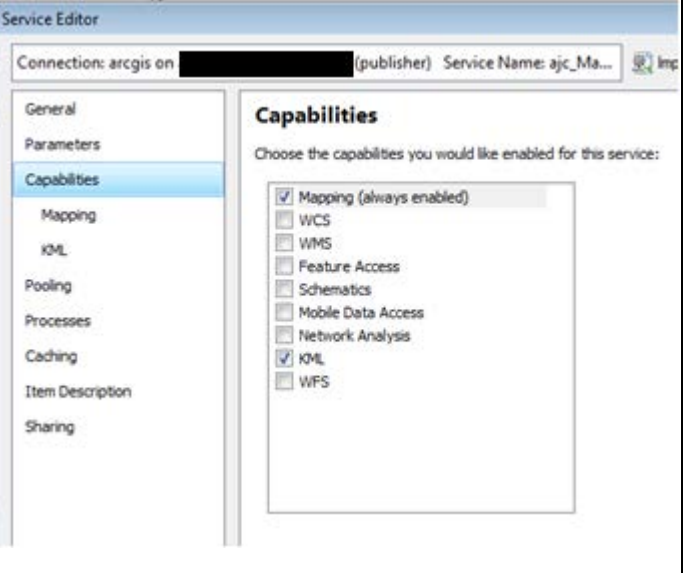
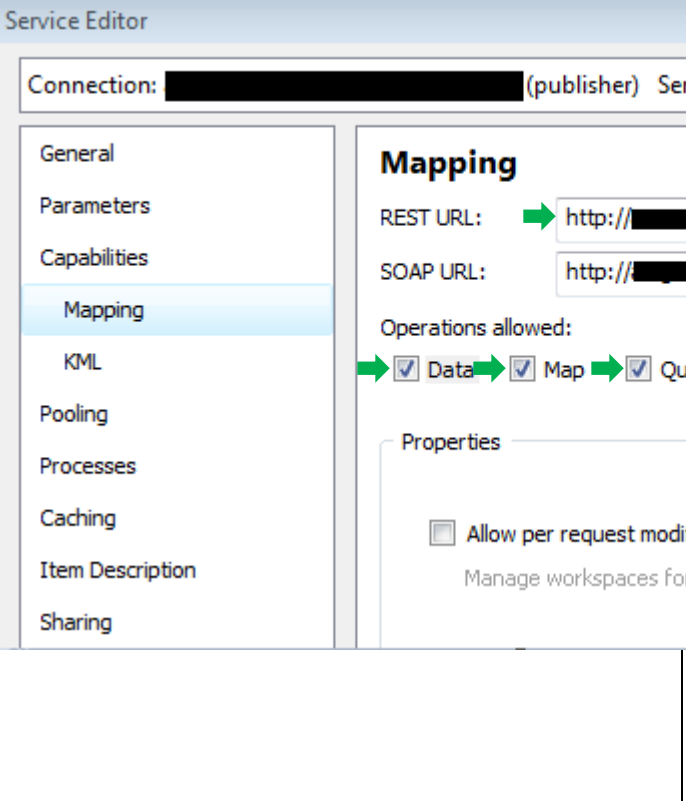
- 1 Open your map document in ArcMap and choose File > Share As > Service from the main menu.

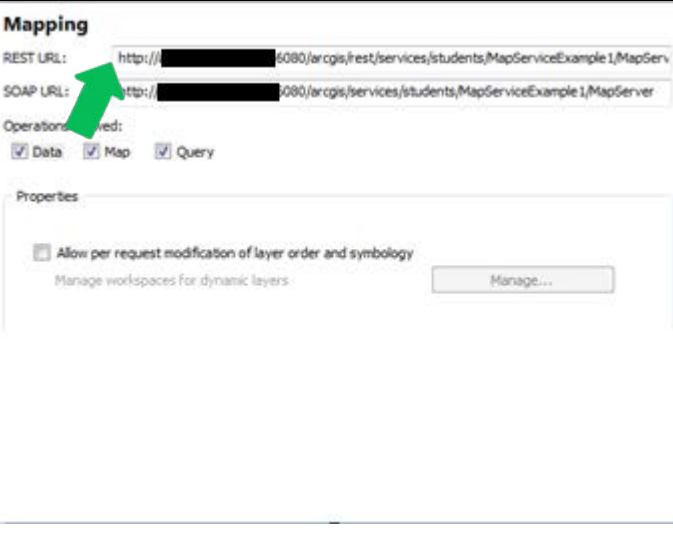
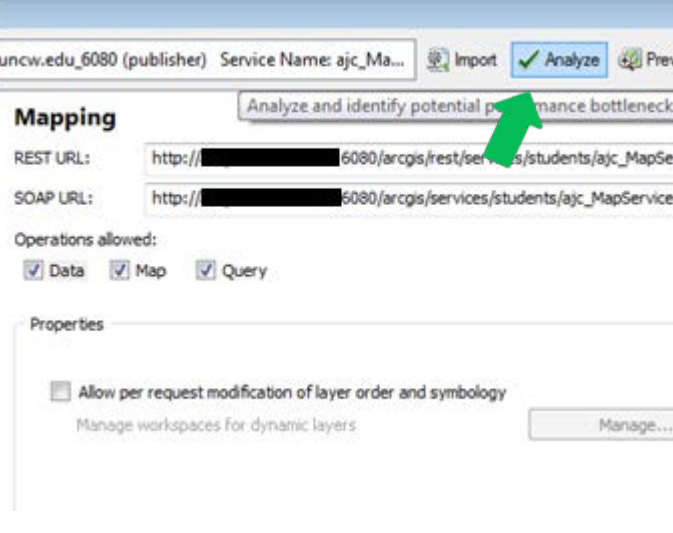
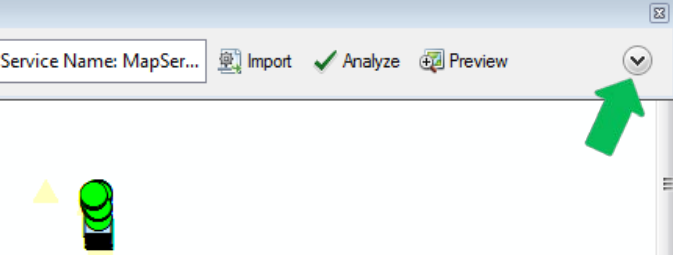


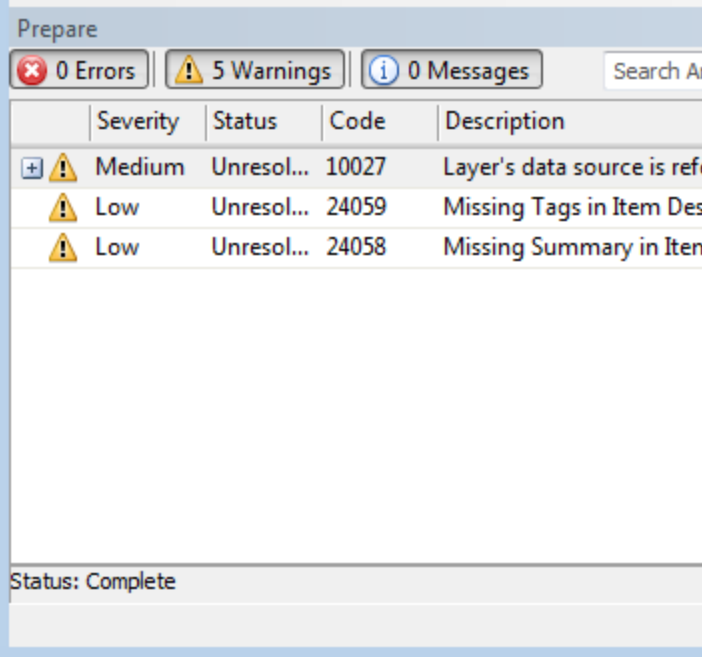

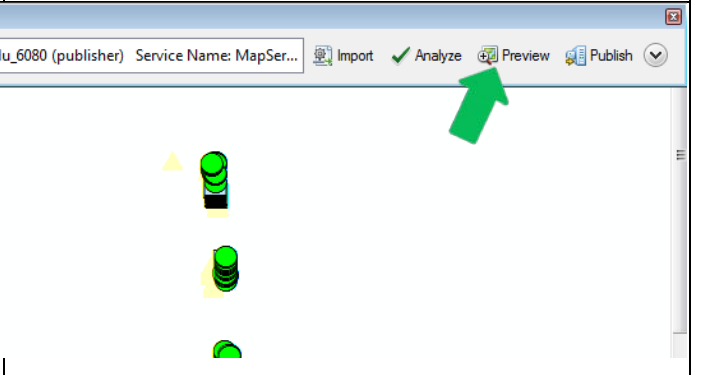
2	In the <i>Share as Service</i> window, choose Publish a service. Click Next.	 <p>The 'Share as Service' dialog box is shown. It features a central graphic of a server with arrows pointing outwards. To the right, there are three radio button options: 'Publish a service' (which is selected), 'Save a service definition file', and 'Overwrite an existing service'. Below these options is a blue hyperlink that reads 'About sharing a service'. At the bottom right, there are two buttons: 'Next >' and 'Cancel'.</p>
3	In the Publish a Service dialog box, click the button to the right of Select an existing connection or create a new one  to create a new connection to the server. If a connection has already been created in the past and appears in the dropdown menu, skip to step 6.	 <p>The 'Publish a Service' dialog box is shown. It has a section titled 'Choose a connection' with a dropdown menu that says 'Select an existing connection or create a new one' and a small 'Add' icon to its right. Below this, the 'Server type:' is set to 'No Connection Set'. There is a 'Service name' field containing the text 'svc_test1'. At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.</p>


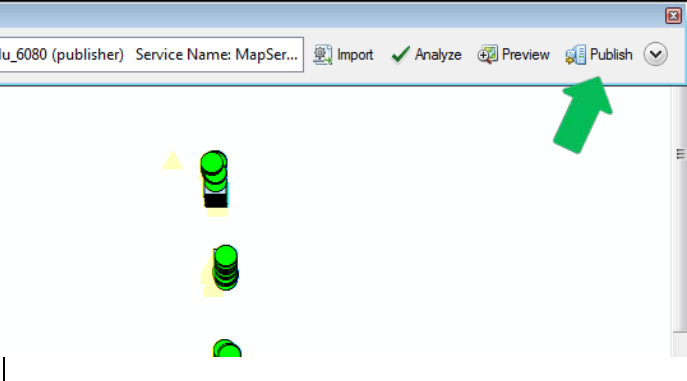
4	<p>Select Publish GIS services</p>	
5	<p>Fill out the following information:</p> <ul style="list-style-type: none"> • Server URL: http:// ####.####.####.####:6080/arcgis • Sever Type: ArcGIS Server (should already be selected) • Staging folder: Keep default • Use ArcGIS Desktop's staging folder: Check (should already be checked) • User name: ##### • Password: ##### • Save Username/Password: Check (should already be checked) 	

6	<p>From the Server Type drop-down list, choose ArcGIS Server, click Next.</p> <p>*placeholder, change service name to include student name?</p>	
7	<p>Select Use existing folder then select the students folder from the dropdown menu. Click Continue.</p>	

8	<p>The <i>Service Editor</i> displays. You'll use the <i>Service Editor</i> to choose what users can do with your map service and take fine-grained control of how the server will expose your service. Click the Capabilities tab.</p>	
9	<p>By default, mapping and KML are enabled. Click Mapping and review the following properties:</p> <p>URL—This is the URL clients use to access the map service. The URL will be formatted as follows: <code>http://###.###.###.###:6080/arcgis/services/<folder name (if applicable)>/<service name>/MapServer</code>.</p> <p>Data—Selecting this option allows client applications to perform attribute searches on the features in your map service.</p> <p>Map—Selecting this option allows client applications to view the map layers in your map service.</p> <p>Query—Selecting this option allows client applications to query the features in your map service.</p>	

10	Copy REST URL to your homework document, you'll be using a slightly modified version of this URL later.	 <p>Mapping</p> <p>REST URL: http://[redacted]:6080/arcgis/rest/services/students/MapServiceExample1/MapServ</p> <p>SOAP URL: http://[redacted]:6080/arcgis/services/students/MapServiceExample1/MapServer</p> <p>Operations allowed:</p> <p><input checked="" type="checkbox"/> Data <input checked="" type="checkbox"/> Map <input checked="" type="checkbox"/> Query</p> <p>Properties</p> <p><input type="checkbox"/> Allow per request modification of layer order and symbology</p> <p>Manage workspaces for dynamic layers: Manage...</p>
11	Click Analyze . This examines your map document to see if it can be published to the server.	 <p>uncw.edu_6080 (publisher) Service Name: ajc_Ma... Import Analyze Preview</p> <p>Mapping Analyze and identify potential performance bottlenecks</p> <p>REST URL: http://[redacted]:6080/arcgis/rest/services/students/ajc_MapSer</p> <p>SOAP URL: http://[redacted]:6080/arcgis/services/students/ajc_MapServiceE</p> <p>Operations allowed:</p> <p><input checked="" type="checkbox"/> Data <input checked="" type="checkbox"/> Map <input checked="" type="checkbox"/> Query</p> <p>Properties</p> <p><input type="checkbox"/> Allow per request modification of layer order and symbology</p> <p>Manage workspaces for dynamic layers: Manage...</p>
	<p>Tip:</p> <p>To give yourself more viewing area when configuring your map service, click the Collapse button at the top-right of the Service Editor.</p>	 <p>Service Name: MapSer... Import Analyze Preview Collapse</p>

12	<p>Fix any Errors in the Prepare window.</p> <p>Double-click on each item in the list to see details about each error message.</p> <p>This must be done before you can publish. You'll likely receive a few warnings such as the following:</p> <ul style="list-style-type: none">• 10027: Layer's data source is referenced via a UNC path• 24059: Missing Tags in Item Description• 24058: Missing Summary in Item Description <p>You can safely disregard these warnings. Remember, you're fixing errors.</p> <p>There is however one warning you need to keep an eye out for. If you receive a warning similar to the following:</p> <p>24011: Data source is not registered with the server and data will be copied to the server</p> <p>Do not proceed. This means your data will be copied over to the server and is not currently in the webgis folder; contact the instructor for next steps.</p>	 <p>Prepare</p> <p>0 Errors 5 Warnings 0 Messages</p> <table><thead><tr><th></th><th>Severity</th><th>Status</th><th>Code</th><th>Description</th></tr></thead><tbody><tr><td>+ !</td><td>Medium</td><td>Unresol...</td><td>10027</td><td>Layer's data source is referenced via a UNC path</td></tr><tr><td>!</td><td>Low</td><td>Unresol...</td><td>24059</td><td>Missing Tags in Item Description</td></tr><tr><td>!</td><td>Low</td><td>Unresol...</td><td>24058</td><td>Missing Summary in Item Description</td></tr></tbody></table> <p>Status: Complete</p>		Severity	Status	Code	Description	+ !	Medium	Unresol...	10027	Layer's data source is referenced via a UNC path	!	Low	Unresol...	24059	Missing Tags in Item Description	!	Low	Unresol...	24058	Missing Summary in Item Description
	Severity	Status	Code	Description																		
+ !	Medium	Unresol...	10027	Layer's data source is referenced via a UNC path																		
!	Low	Unresol...	24059	Missing Tags in Item Description																		
!	Low	Unresol...	24058	Missing Summary in Item Description																		
13	<p>Optionally, in the <i>Service Editor</i>, click Preview . This can give you an idea of how your map will look when viewed on the web. See <i>Previewing your map</i> for more information.</p>	 <p>lu_6080 (publisher) Service Name: MapSer...</p> <p>Import Analyze Preview Publish</p>																				

14	Once you've verified there are no errors in your map document, click Publish  .	
15	When your service has been successfully published, notify the instructor so he/she can “unlock” the service for public viewing. Once this is done you can view your published service at the REST URL copied in step 10.	

APPENDIX Q: WEB MAP CREATION INSTRUCTIONS

Create a Web Map

Modified and adapted from Esri's instructions* for use by UNCW Department of Geography & Geology

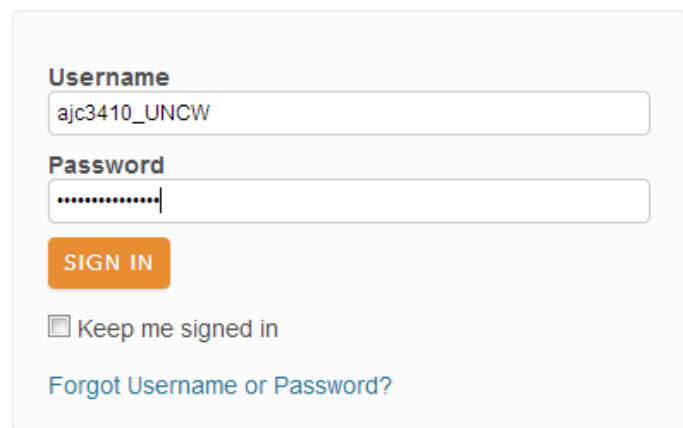
Prerequisite checklist:

- Have you created a service (or services)?
 - Do you have their URLs? (usually: `http://###.###.###.###:6080/arcgis/rest/services/<folder_name>/<service_name>/MapServer`)
 - Has the instructor “unlocked” the service using the ArcGIS Server Manager, making it public?
- Has the instructor created an ArcGIS.com online account for you?

1 Login to ArcGIS.com
Open your browser and go the following URL:
<https://#####.arcgis.com/home/signin.html>

2 Log in using your assigned User ID, if this is your first time logging in you may be prompted to change your password.

Sign in to University of North Carolina Wilmington



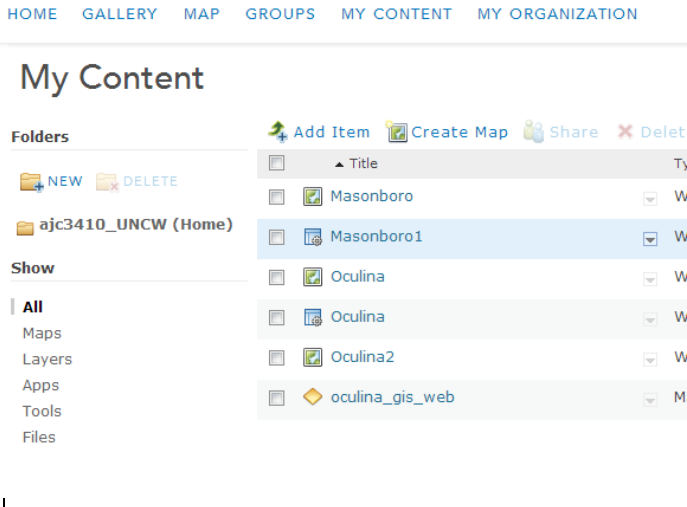
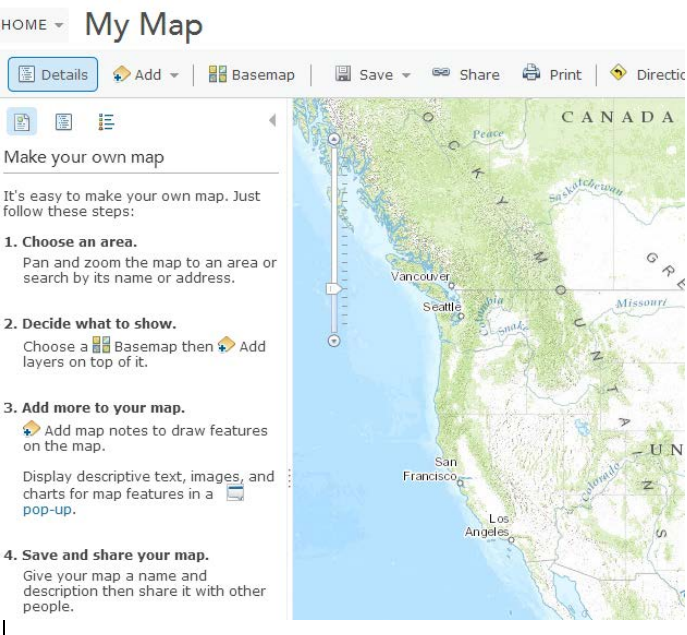
Username
ajc3410_UNCW

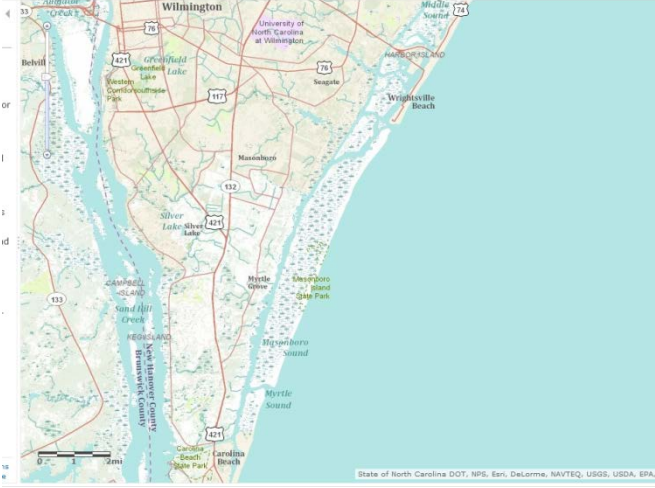
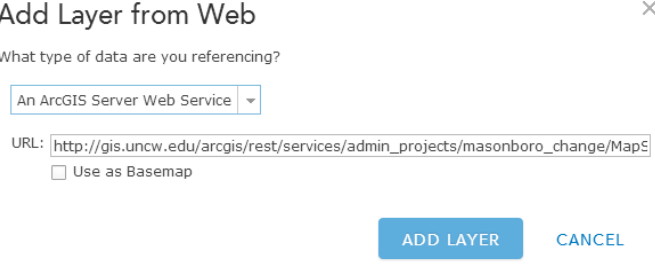
Password
.....

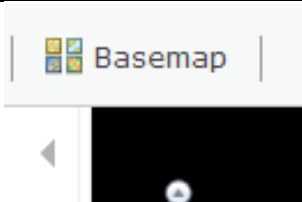
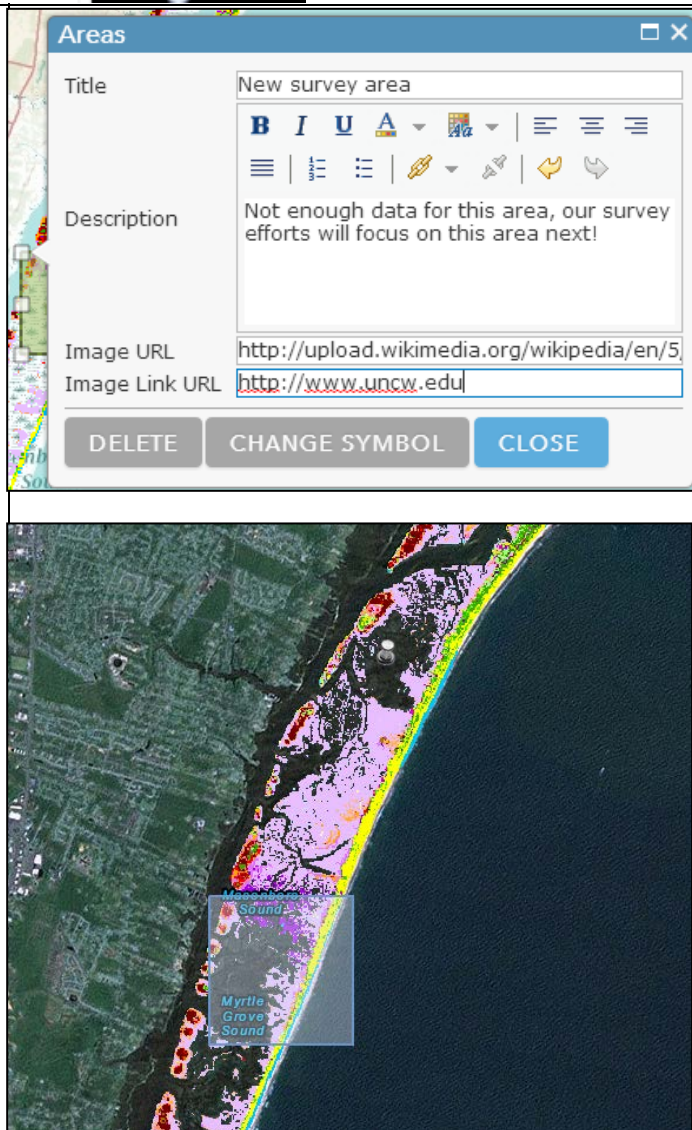
SIGN IN

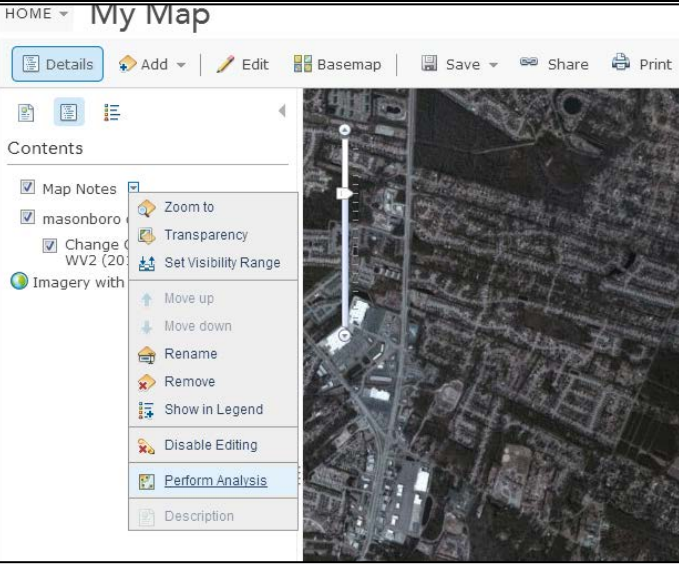
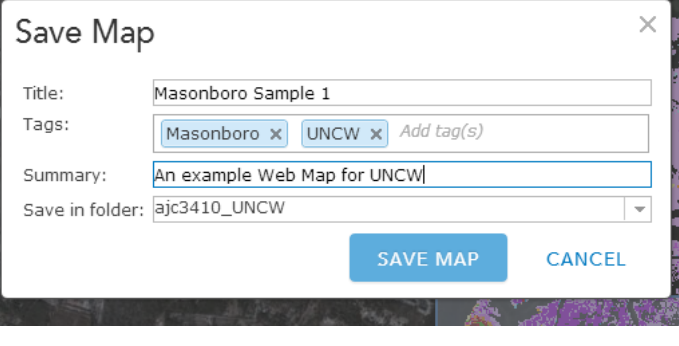
☐ Keep me signed in

[Forgot Username or Password?](#)

3	At the top of the screen, click on “MAP”	
4	From this screen you can start building your Web Map.	
5	Follow the instructions they’ve given you:	

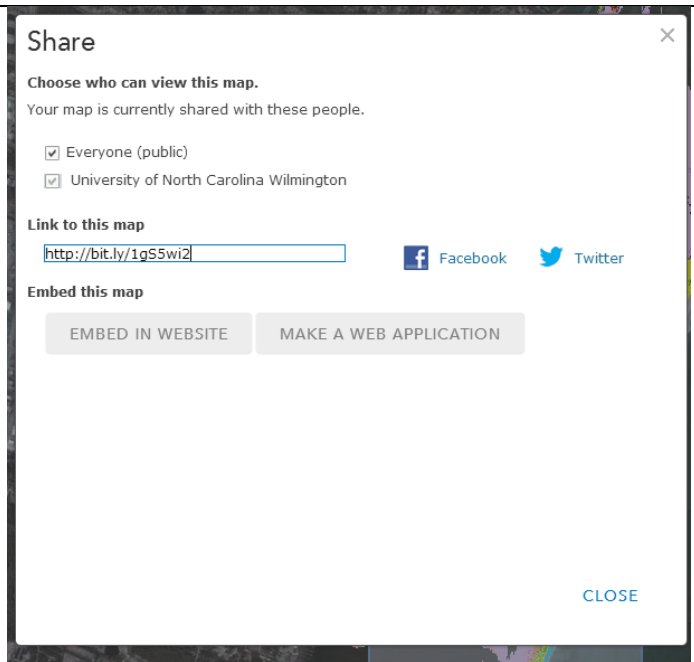
5a	<p>Choose an area: In this example I'm using Masonboro data so I've zoomed in to that location</p>	
5b	<p>Decide what to show: Here's where you add service URLs and any other online data you'd like. To do this select Add > Add layers from web. Select "An ArcGIS Web Server Service" and then input a modified version of your service URL.</p> <p>Earlier you should've copied a REST URL for your service that looked something like this: <code>http://###.###.###.###:6080/arcgis/rest/services/<folder_name>/<project_name>/MapServer</code></p> <p>You need to modify this URL slightly, actually you just need to modify the first part of it. Instead of <code>http://###.###.###.###:6080</code>, type <code>http://gis.uncw.edu</code>, then add on the rest of the URL. For example here's what I'm adding: <code>http://gis.uncw.edu/arcgis/rest/services/admin_projects/masonboro_change/MapServer</code></p> <p>If you're interested in why you had to change the URL, it's because ArcGIS.com has to access your service through a machine that we've made publicly visible (not many machines on campus have this capability).</p> <p>Repeat this step for as many services</p>	

	<p>(or other data) you wish to add!</p> <p>*This one isn't a basemap so I won't be checking the box at the bottom. If it is a basemap, check the box.</p>	
5c	<p>Click on Basemap at the top if you'd like to change the default basemap.</p>	
5d	<p>Get some experience adding notes and symbols to the WebMap. To the left of the Basemap button is an Edit button, click this. Add any symbols or shapes you think might help your map along with a title and notes for each. These will appear in a popup when the user clicks on them. You can change the color and specific symbol you've selected by clicking "Change Symbol" once you've added the feature. Here I've added a pushpin and also drawn a rectangle. You can do something similar and add your own descriptions. Whatever picture and link you provide will also appear when the user clicks on this area, these aren't required but occasionally can be useful. I've included a URL for the UNCW logo, when users click on it, it will take them to http://www.uncw.edu</p>	

5e	<p>Depending on the service data you've added, there are also a variety of analysis operations available when you draw shapes. Though it's beyond the scope of this lesson, once you've finished drawing a rectangle, click the arrow to the right of your notes layer and go to "Perform Analysis". There are many available tools for analysis here.</p>	
5f	<p>Some layers can also be enhanced through use of pop-ups. More data can be found about adding pop-ups to compatible layers here: http://resources.arcgis.com/en/help/arcgisonline/index.html#/010q0000004m000000</p>	
6	<p>Finally, save your map by clicking on the "Save" button. Choose a title, a couple of tags and a summary.</p>	

7

You can share your map a variety of ways. Talk to your instructor about the best way to share it. If you're planning on building an application, go ahead and click on "Make A Web Application" and continue with the ArcGIS.com Web Application instructions. If you'd like to see what your WebMap looks like, copy and paste the "Link to this map" URL into a new tab.



APPENDIX R: INSTRUCTIONS FOR ARCGIS.COM APPLICATION CREATION

Create an ArcGIS.com Online Map Application

Prerequisite checklist:

- Have you created a web map on ArcGIS.com?

1 If you're picking up from where "Create a Web Map" left off, in the Share window, click "Make a Web Application".

If you have a saved Web Map, log into ArcGIS.com and navigate to your content. Open your web map and click "Share", this will get you to the same screen.

Share

Choose who can view this map.

Your map is currently shared with these people.

- ☒ Everyone (public)
- ☒ University of North Carolina Wilmington

Link to this map

<http://bit.ly/1gS6WZX>


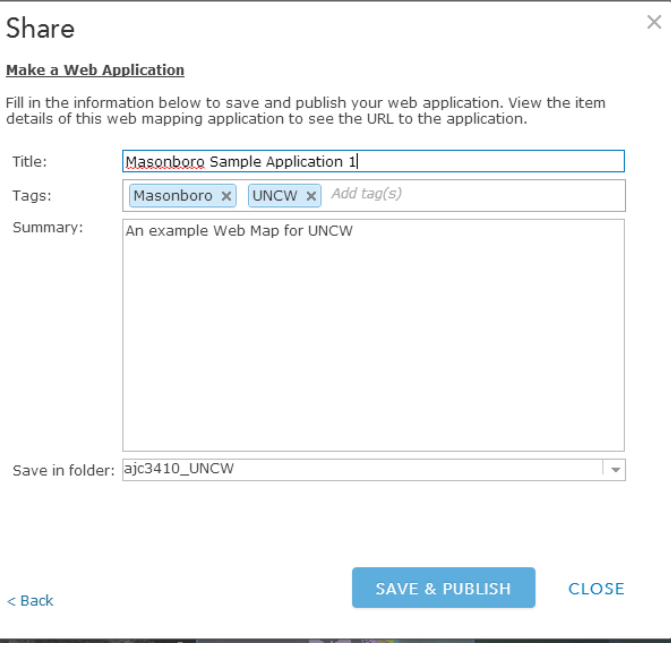


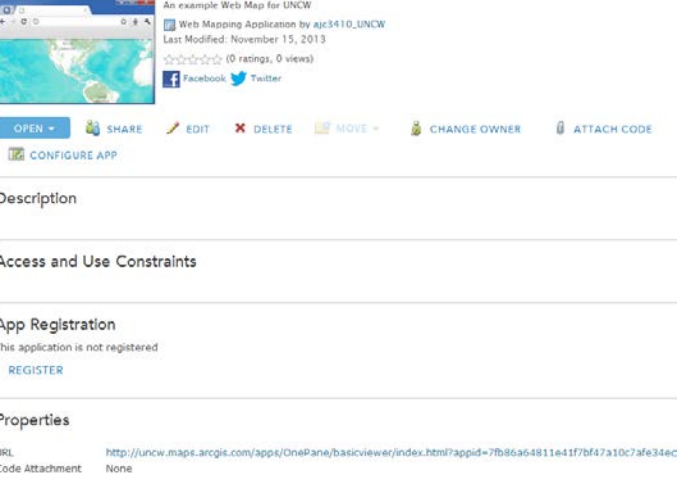

Embed this map


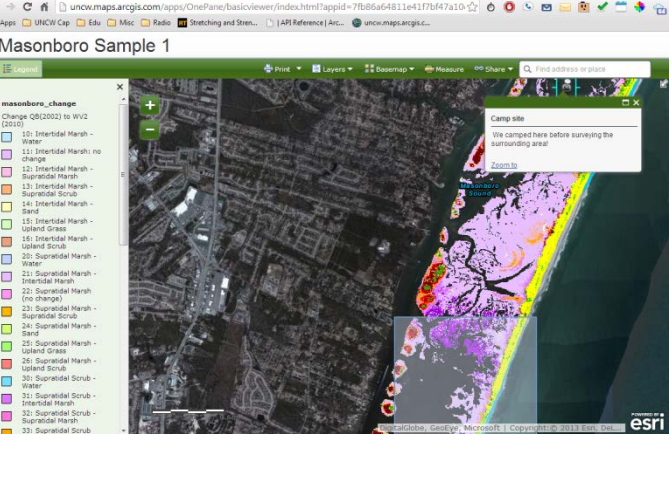
EMBED IN WEBSITE

MAKE A WEB APPLICATION

CLOSE

2	<p>Next you'll be given several application template options. For this exercise, it's simplest to choose the "Basic Viewer" option. Choose this then click "Publish".</p> <p>If you find you need greater flexibility and customization options and don't mind writing some JavaScript code, refer instead to "Create a UNCW Server-based Application".</p>	
3	<p>Choose a title for your application, the tags and summary should be auto-populated from your web map. You may choose to change these or just accept them as they are. Click "Save & Publish".</p>	

3	<p>You should get a window that indicates your Application has been successfully published. Click “go to the item now” to configure and share the application.</p>	<h3>Share</h3> <p>Make a Web Application</p> <p>You have successfully published your web application. Here are a few things to do before other people can use it.</p> <p>Share your application - The application is only viewable by you until you share it with others. Just like you shared the web map this application references, you need to share this application as well. Typically, you'll share it with the same people you shared the web map with.</p> <p>Configure your application - If the application can be configured, select the 'Configure App' button on the item details page to change how the application looks and works.</p> <p>You can go to the item now to share and configure your application or click Close below to return to the map.</p>
4	<p>Next you'll be brought to a screen where you can open or configure your application. Click the “Configure App” link.</p>	<h3>Masonboro Sample Application 1</h3>  <p>Description</p> <p>Access and Use Constraints</p> <p>App Registration This application is not registered REGISTER</p> <p>Properties URL: http://uncw.maps.arcgis.com/apps/OnePane/basicviewer/index.html?appid=7fb86a64811e41f7bf47a10c7afe34ec Code Attachment: None</p>
5	<p>You have several options here. You can choose color settings and various other configurable items. Feel free to include or eliminate any items you'd like. You can always change this later. In my example I've kept most of the defaults but changed the color scheme to Green and configured the print settings to include the legend on the printout.</p> <p>When you're done click save, it should show you your map with changes implemented. If you don't like your changes, just change them. When you're done, click cancel to go back to the Application dashboard.</p>	 <p>General Settings</p> <p>Color Scheme: Green</p> <p><input checked="" type="checkbox"/> Show Title Title Text: Defaults to map name</p> <p>Logo on map: URL to image</p> <p><input checked="" type="checkbox"/> Include Overview Map</p> <p>Menu Items</p> <p><input checked="" type="checkbox"/> Legend * <input checked="" type="checkbox"/> Details * <input checked="" type="checkbox"/> Editor * <input checked="" type="checkbox"/> Time Slider * <input checked="" type="checkbox"/> Print <input checked="" type="checkbox"/> Layer List * <input checked="" type="checkbox"/> Basemaps <input checked="" type="checkbox"/> Bookmarks <input checked="" type="checkbox"/> Measure <input checked="" type="checkbox"/> Share <input type="checkbox"/> Elevation <input type="checkbox"/> Show Elevation Difference <input checked="" type="checkbox"/> Search <input type="checkbox"/> Search for locations within current extent</p> <p>* These menu items will appear in the application when the web map has layers that require them.</p> <p>Print Settings</p> <p><input checked="" type="checkbox"/> Display Legend on Printout Layout: Format: </p>

		 <p>The screenshot shows the ArcGIS OnePane application interface. On the left, there's a search bar with the text "Find address or place". Below it, a map of Masonboro Sound is displayed with various land use layers. The map is overlaid with a grid of colored polygons representing different land use categories. On the right, there's a "Menu Items" section with a list of tools: Legend, Details, Editor, Time Slider, Print, Layer List, Basemaps, Bookmarks, Measure, Share, Elevation, Show Elevation Difference, Search, and a checkbox for "Search for locations within current extent". Below this is a "Print Settings" section with a checkbox for "Display Legend on Printout", a "Layout:" dropdown, and a "Format:" dropdown. At the bottom right, there are "SAVE", "RESET", and "CANCEL" buttons.</p>
6	<p>At this point you can open and view your final application (click Open > View Application). The application comes bundled with several tools; feel free to test each one. If you made your site visible to the public, the URL you're using to view the application is the same one you can use to view it from anywhere! For example, the quick sample application used for the tutorial is located here:</p> <p>http://####.arcgis.com/apps/OnePane/basicviewer/index.html?appid=7fb86a64811e41f7bf47a10c7afe34ed</p>	 <p>The screenshot shows the ArcGIS OnePane application interface. On the left, there's a search bar with the text "Find address or place". Below it, a map of Masonboro Sound is displayed with various land use layers. The map is overlaid with a grid of colored polygons representing different land use categories. On the right, there's a "Menu Items" section with a list of tools: Legend, Details, Editor, Time Slider, Print, Layer List, Basemaps, Bookmarks, Measure, Share, Elevation, Show Elevation Difference, Search, and a checkbox for "Search for locations within current extent". Below this is a "Print Settings" section with a checkbox for "Display Legend on Printout", a "Layout:" dropdown, and a "Format:" dropdown. At the bottom right, there are "SAVE", "RESET", and "CANCEL" buttons.</p>

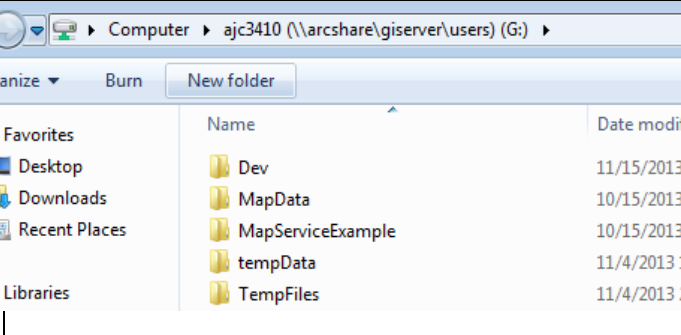
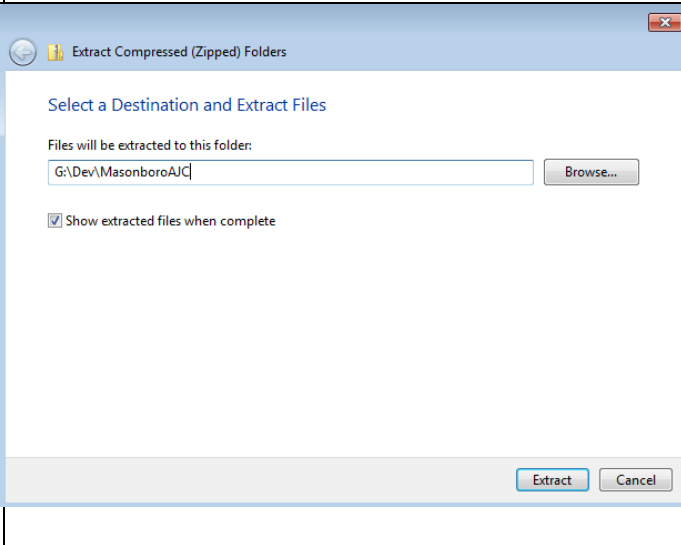
APPENDIX S: INSTRUCTIONS FOR JAVASCRIPT APPLICATION CREATION FROM A TEMPLATE

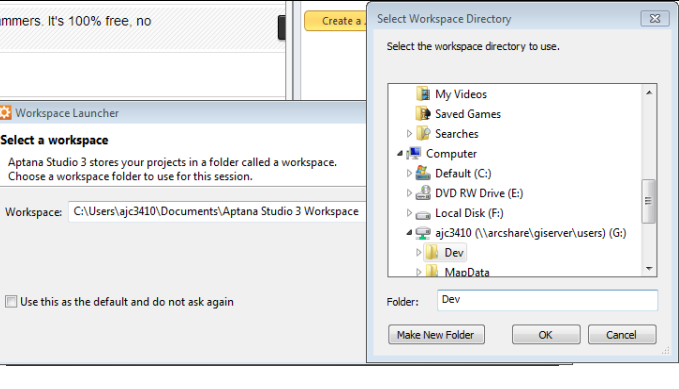
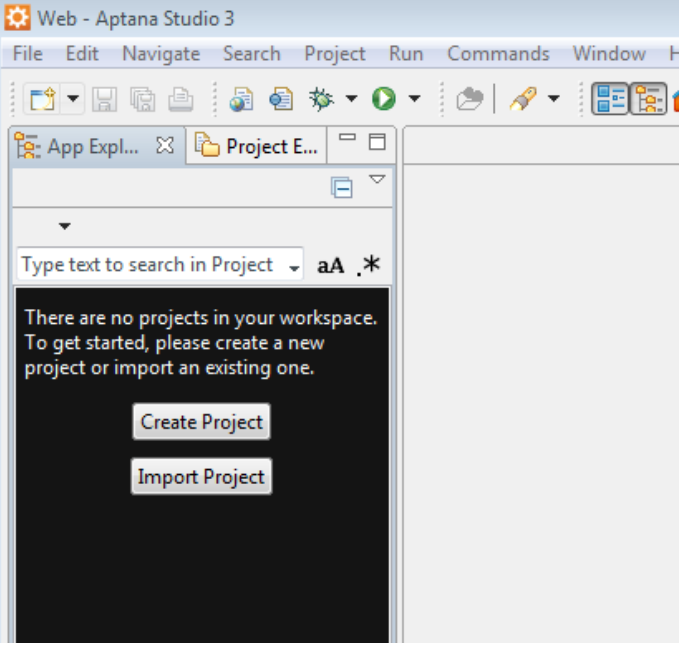
Create a Javascript Map Application from a Template

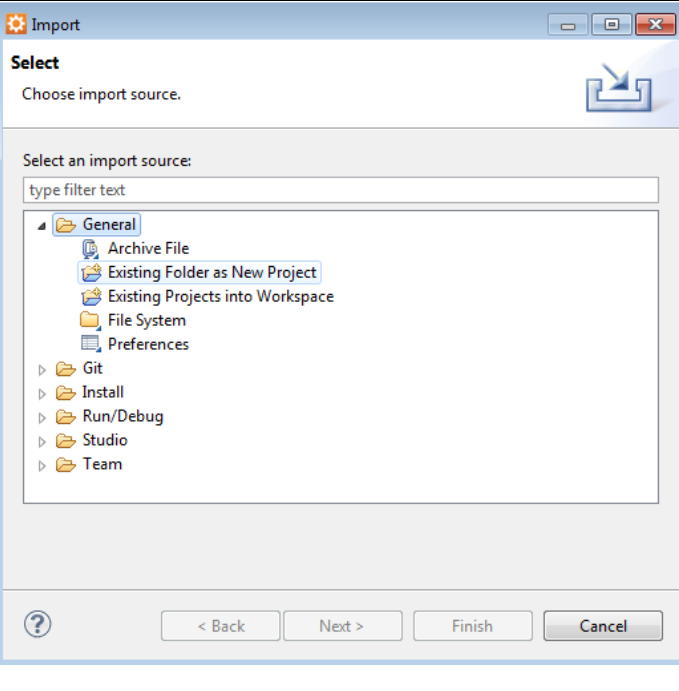
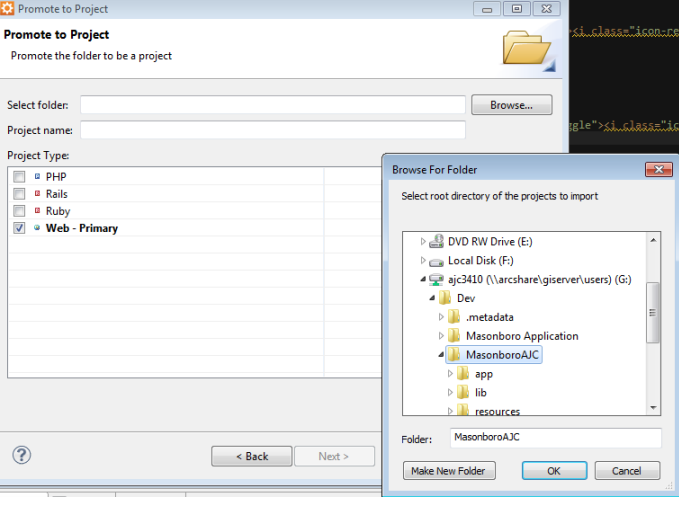
Prerequisite checklist:

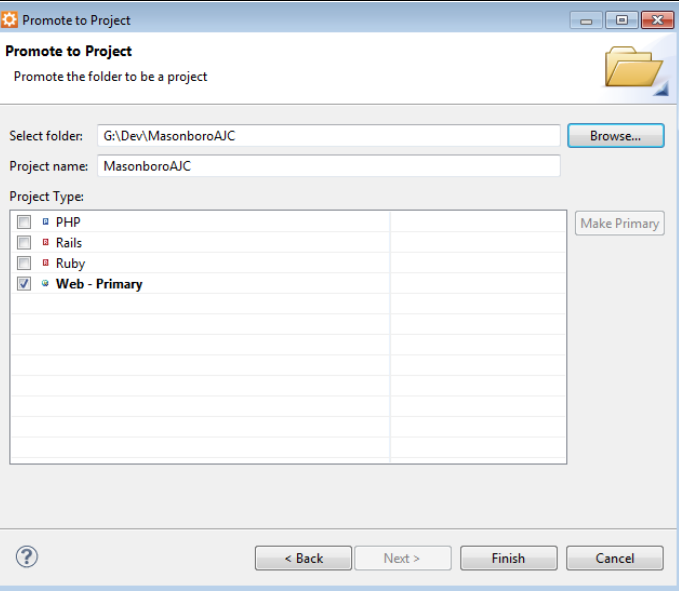
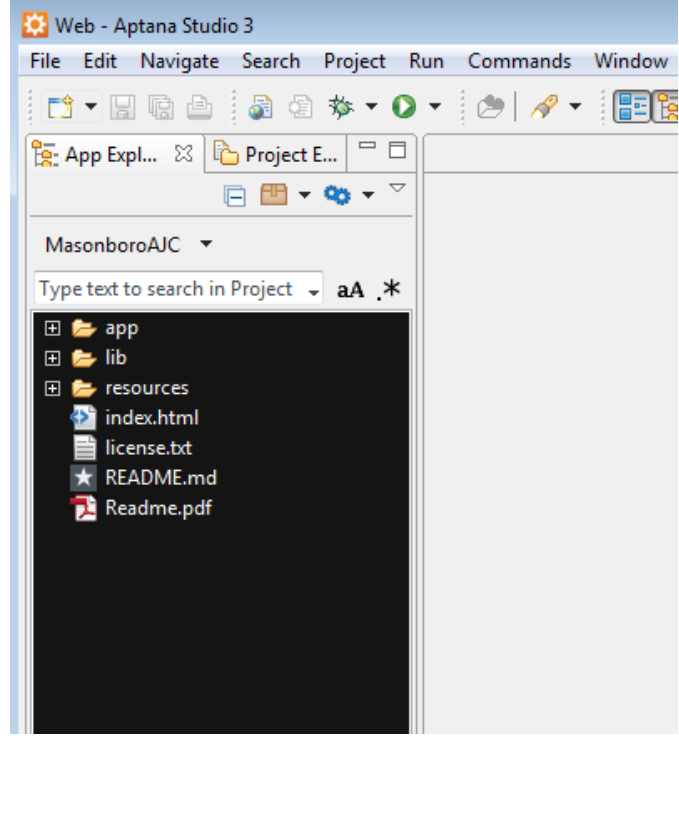
- Have you created a web map (or web maps) on ArcGIS.com?
- The template used for this instruction is primarily targeted


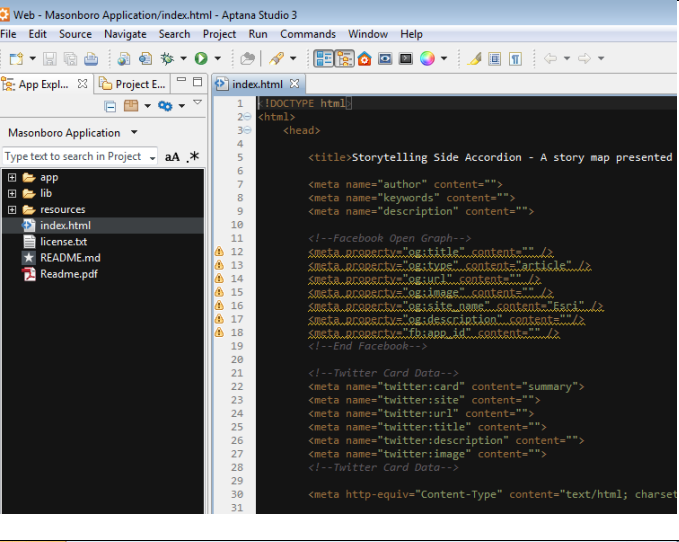

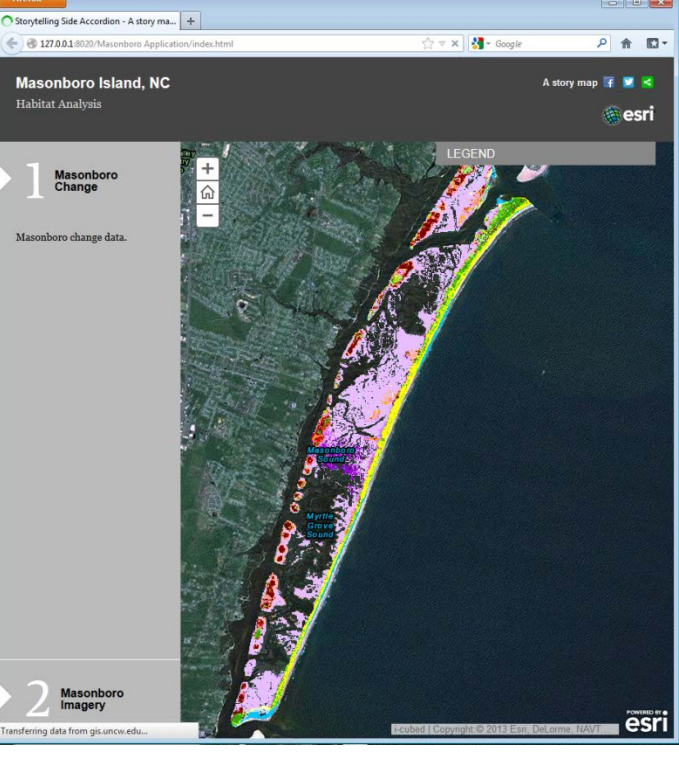
A basic sample of this template has been placed here: <http://gis.uncw.edu/masonboro/>. You may use it for reference.

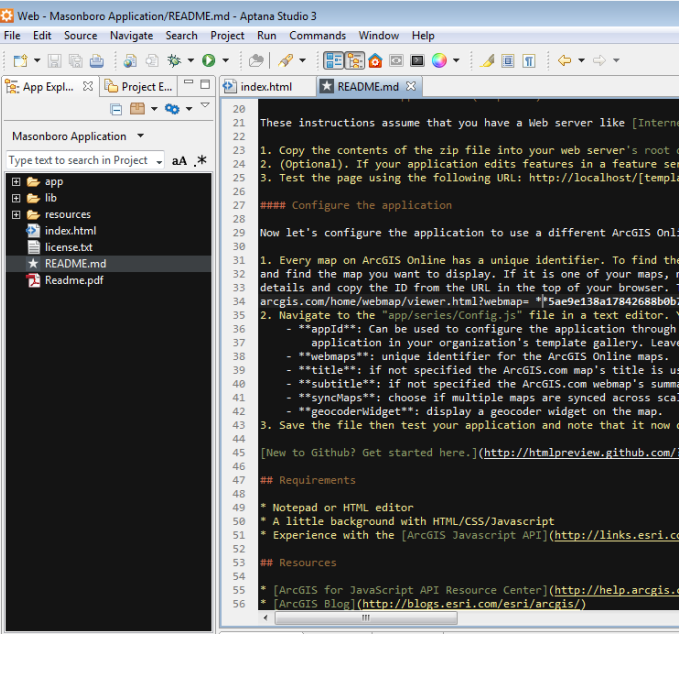
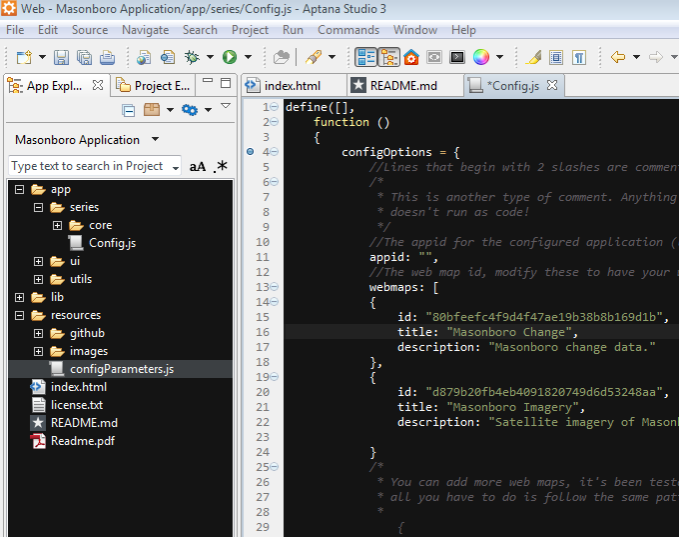
1	Create a folder on your G drive called 'Dev' (G:\Dev).	
2	<p>Navigate to the following folder: %webgis% \application_templates</p> <p>Copy 'Masonboro Side-accordion Template.zip' to the Dev folder you created above. Once you've copied it over, go to your Dev folder, right-click the zip file and extract it there. This will create a folder with the same name there. Change the name to 'Masonboro<your initials>'. For example on my application this line would read: G:\Dev\MasonboroAJC</p>	

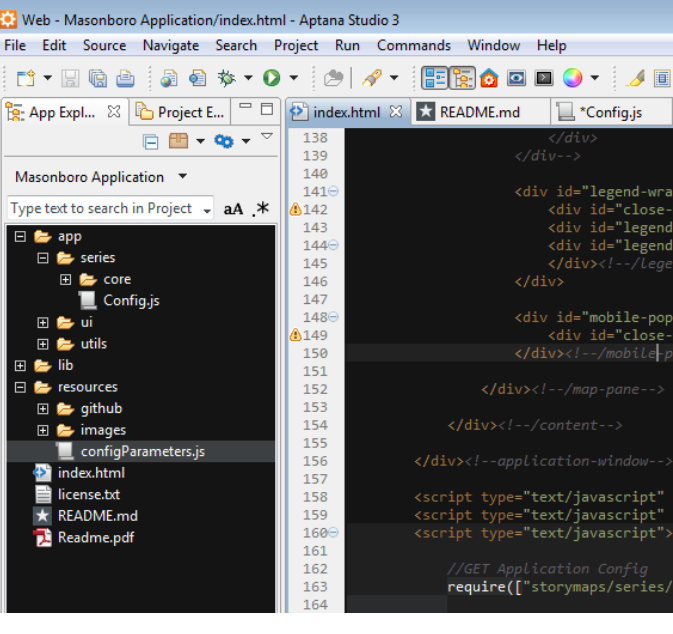
	<p>Launch Aptana (Start Menu > All Programs > Aptana > Aptana Studio 3). You'll be prompted to select a Workspace, browse to G:\Dev. Click ok.</p>	
4	<p>Aptana will then launch. On the left screen will be a prompt to either create a project or import a project. Since we're using a template, click 'Import Project'.</p>	

5	Expand 'General' and select 'Existing Folder as New Project'	
6	Browse to the Masonboro<your initials> folder you just created and click 'Ok'.	

7	<p>Web – Primary should be selected by default but if it's not, go ahead and select it. Click 'Finish'.</p>	
8	<p>On the left hand side of Aptana, it should look similar to the image here. Double-click on 'index.html' to load it up in the main window.</p>	

9	<p>If there are any warnings, don't worry, it's normal. Without modifying anything, go ahead and click the green play button  at the top.</p>	
10	<p>This should load the application into your browser. Any time you want to run your application locally to preview how it will appear once it's been published, click the green play button. To stop it you can either close your browser or click the stop button  on the lower right of the Aptana window.</p>	

11	<p>Now open 'README.MD'. This file has useful information for configuring your application. In particular, focus on the instructions under:</p> <pre>#### Configure the application</pre> <p>In summary, these tell you to:</p> <ol style="list-style-type: none"> 1. Understand that your ArcGIS.com web maps have unique identifiers. You'll need these. 2. Open app/series/Config.js 3. Save everything any time you make any changes 	
12	<p>Go ahead and open app/series/Config.js now. There are instructions in this file, in the form of code comments. Between these instructions and the instructions provided in README.md, modify the code IN CONFIG.JS ONLY to customize your application. If you're very familiar with web development, you're welcome to modify other files in this application but this can be very tricky. You've been warned!</p>	

13	<p>Any time you want to run the application to test your changes, go back to index.html then click the play button like you did before. This is important since Aptana executes the application from the current window you're focused on (index.html is the home page of the application). In short: modify Config.js, run application from index.html.</p>	 <p>The screenshot shows the Aptana Studio 3 interface. The top menu bar includes File, Edit, Source, Navigate, Search, Project, Run, Commands, Window, and Help. The left sidebar displays the 'Masonboro Application' project structure with folders like app, series, core, ui, utils, lib, resources, github, and images, along with files like index.html, license.txt, README.md, and Readme.pdf. The main editor window shows the 'index.html' file, which contains HTML code for a web application, including a legend, a map pane, and a content area. The code is partially visible, showing tags like <div id='legend-wrap', <div id='close-m', <div id='legend', <div id='legend', </div><!--/Legen, <div id='mobile-popu', <div id='close-m', </div><!--/mobile-p, </div><!--/map-pane-->, </div><!--/content-->, </div><!--/application-window-->, <script type='text/javascript' s, <script type='text/javascript' s, <script type='text/javascript'>, //GET Application Config, and require(['storymaps/series/</p>
14	<p>If, after modifying your application you'd like the professor to publish your completed application on the public web server, do the following.</p> <ol style="list-style-type: none"> 1. Discuss it with the professor first. Ensure there's no further work required prior to publishing. 2. If you've made it to this point, there are several parts of your project you'll want the professor to have access to. You should have created your own service(s), created web maps on ArcGIS.com and now you've just created an application that uses these. <ol style="list-style-type: none"> a. Make sure any .mxd files used to create the original services you're using have been saved in the webgis folder in the subfolder designated by the professor. b. Send an email to the professor with the following information: <ol style="list-style-type: none"> i. Path to the .mxd file(s) in the sub-folder within webgis ii. Names of Web Maps used in this application c. Once the professor posts the application, it should be available at the following address: <a href="http://gis.uncw.edu/masonboro<your initials>/">http://gis.uncw.edu/masonboro<your initials>/ 	

APPENDIX T: TOOLS USED ON THIS PROJECT

Tool/Application	Analysis & Design	Sample Application	Service & Application Migration	Web Portal Update
Microsoft© Windows™ 7 64-Bit OS	Y	Y	Y	Y
Google© Chrome™ Version 29.0.x	Y	Y	Y	Y
Microsoft© Internet Explorer versions 9 and 10	N	Y	Y	Y
Mozilla Firefox version 17.0.x	N	Y	Y	Y
Cisco© AnyConnect™ VPN Client	Y	Y	Y	Y
ESRI© ArcGIS Server™ Enterprise Edition 10.1	Y	Y	Y	N
ESRI© ArcGIS Server™ Enterprise Edition 10.0	Y	N	N	N
ESRI© ArcGIS Server™ Enterprise Edition 9.2	Y	N	Y	N
ESRI© ArcGIS Desktop™	N	Y	Y	N
Notepad++ Version 6.3.x	N	Y	Y	Y

Aptana Version 3.0	N	Y	Y	N
Oracle© Java™ Platform SE 7u7	N	Y	Y	N
CrystalDiskInfo 5.6.2	Y	N	N	N
Cygwin 1.7.22 (for git version control)	N	Y	Y	N

APPENDIX U: LOGGING INTO ARCGIS SERVER 10.1 AS ADMIN

The quick way:

1. If not already logged in, log in to the UNCW network and go to the following URL: `http://###.###.###.###:6080/arcgis/manager/login.html?url=`
2. Enter User name: #####
3. Password: #####

The longer way (gives you access to the server and machine if you need it):

1. If not already logged in, log in to the UNCW network.
2. On your computer, go to : Start > All Programs > Accessories > Remote Desktop Connection
3. Type in the server name: `###.###.###.###`
4. It will prompt you for your user name: `UNCW\<UNCW_user_name>`
5. Enter password
6. Click OK on any other popups it gives you.
7. Once logged in, go to Start > All Programs > ArcGIS > ArcGIS 10.1 for Server > Manager
8. Enter User name: #####
9. Password: #####

APPENDIX V: BEST PRACTICES

Services:

- **Saving Data:** Map data for services should be stored in a subfolder in the webgis folder with the mxd file. It's never a bad idea to keep a backup of this data on your personal campus drive or a thumb drive.
- **Naming Services:** Often there will be multiple individuals creating services describing a similar area (for example, "Masonboro"). Using lowercase letters, name your service using your initials, a dash, then a good, simple, descriptive name for the service. Something like: "jfk-masonboro".
- **# of Services:** Try to only create one service. You can create multiple web maps and applications later that only use certain layers in the service if you'd like. You can also turn layers on and off in web maps and applications.

JavaScript Applications:

- **Saving work:** Since the concept of 'source control' is beyond the scope of current application development, it's important to save and back-up your work frequently!
- **Using new templates:** Esri has many different templates available for download and customization. These templates can allow a person to produce some very advanced applications with relatively little work. In order to use them, it's often necessary to tailor your data to match the template's requirements. For example, the template might require a .csv file (a simple spreadsheet) populated with data and placed in a certain folder. Find the best template for your project, tailor your data as it instructs you to and you should be on your way to a very nice application.
- **Writing custom JavaScript:** It's certainly possible to step outside of the boundaries of what you're supposed to modify in order to add widgets and further customize your application. If you choose to try this, please save your application and create a copy first. Also, understand that an advanced understanding of JavaScript and HTML went into creating this template; it takes a similar level of knowledge to modify it!

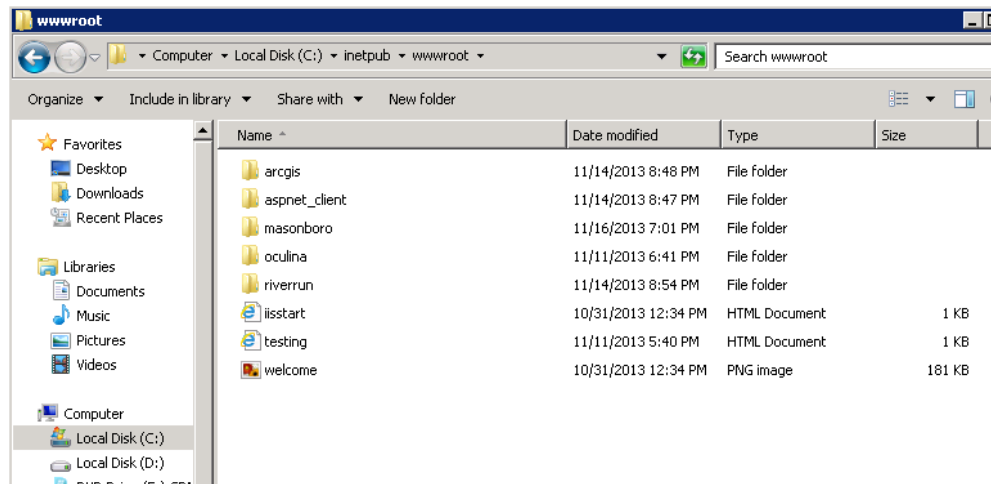
Administration:

- **Services:**
 - **Web gis folder maintenance:** Since the web gis folder is where all service data will be stored, it may be a good idea to sub-divide the folder and instruct individual courses and/or people on where to place their data.
 - **Service fragility:** It's very likely that services will need to be updated or modified down the road. For services you want to preserve and work with in the future, always make sure you have the original .mxd file.
- **ArcGIS.com:**
 - **Project maintenance:** Periodically it will be necessary to remove old accounts and to create new ones. You may want to create folders on your account so that you can transfer old student web maps and applications that are hosted on ArcGIS.com to your account.
 - **Updating projects owned by others:** As an administrator, you can view projects and change titles and basic metadata. To make more advanced changes, you need to change ownership of the web map or application to yourself. You can then reassign it back to the user if needed.
- **JavaScript Applications:**
 - **IIS 7 (Web Server) Publishing:** The new web server is publicly-facing, so it's important that only you and certain administrators have access to it. Anything put in the folder "C:\inetpub\wwwroot\" will be visible to the public. A good process for publishing student applications might be:
 - i. Student creates application
 - ii. Student emails professor zipped application
 - iii. Professor reviews application in Aptana or other IDE
 - iv. If the application requires no further modification, professor publishes application to web server

APPENDIX W: DEPLOYING JAVASCRIPT APPLICATIONS

Prerequisite: User should be a server administrator for //server03, if this is not already the case and you require it, contact ITSD.

1. If not already logged in, log in to the UNCW network.
2. Copy the folder containing the JavaScript project you wish to publish to your shared network folder. On your UNCW connected PC, this usually looks something like: \\arcshare\giserver\users
3. On your computer, go to : Start > All Programs > Accessories > Remote Desktop Connection
4. For the Computer field, type the IP address (for \\server03)
5. It will prompt you for your user name: UNCW\<UNCW_user_name>
6. Enter password
7. Click OK on any other popups it gives you.
8. Once logged in, on the server, open two folders: the shared folder referenced in step 2 and the folder: c:\inetpub\wwwroot



9. You'll see other projects in this folder which are currently published to the web server.
10. To publish your new application, copy the folder containing the new application from your shared folder to the wwwroot folder. The folder name will be part of the URL so make sure it's lower case and doesn't have any special characters. After this, the new JavaScript application should be viewable to the public at the URL: http://gis.uncw.edu/<application_name>