

# **THE CHALLENGES OF BUILDING AN ENTERPRISE-QUALITY WEB MAPPING SOLUTION FOR STATE GOVERNMENT**

Dixie Davis  
Chris Judd

## **ABSTRACT**

This paper discusses the lessons learned in designing, building and implementing a highly available, fault-tolerant and distributed ArcIMS system. Topics addressed include: using a Reverse Proxy Server for enhanced security, Server Load Balancing, configuring multiple Spatial Servers, ArcIMS Fail-over and the development of remote access procedures for an in-house development community. The challenge was to build a commercial-grade, centrally-managed, web mapping system that would replace an existing handful of independent, internet and intranet map servers.

## **1 INTRODUCTION**

The Florida Department of Environmental Protection (FDEP) is a State agency of approximately 5000 employees safeguarding Florida's environment through its "More Protection, Less Process" mission. Since the late eighties, Department scientists, engineers and managers have employed GIS as a planning, analysis and decision-making tool. Beginning with a focused few within the Division of Water Resource Management (formerly Division of Water Facilities), the GIS community at FDEP now numbers over 350. Evolving from the lab and technical arenas, GIS is currently involved in all Departmental operations from regulatory to IT. Pockets of GIS expertise and support can now be found throughout the Agency's offices from core staff in Tallahassee to personnel located in District, Satellite and Park offices across the State.

Centralized GIS support comes from the Bureau of Information Systems' GIS Section whose staff is nominally charged with supporting the Department's geographic information needs. It is from this group and from GIS support staff within the Division of Water Resource Management (WRM) that the first offerings of ArcIMS websites appeared in early 2001. These early sites, maintained separately by the BIS GIS Section and the WRM GIS team, were hosted on independent, single servers that performed all web-serving, map-serving and data-serving tasks. Knowledge-transfer and trouble-shooting efforts between the two groups were hampered because the sites and servers shared little in common beyond the fact that the two machines were both running ArcIMS 3.1 and were serving predominantly shapefile-based data. This disparate development diluted the Department's overall ArcIMS focus for a time and threatened to frustrate the advancement of ArcIMS throughout the Agency.

Fortunately, within and without the Department, a nexus of opportunities and contingencies occurred culminating in late 2002 which when linked produced the environment that fostered and facilitated the cooperative growth, department-wide use and maturation of ArcIMS. These developments included: the pervasive use of the

Internet; Network infrastructure improvements; advancements in database technologies; elevated learning curves for standard GIS software; tight State budgets forcing continued pressure to return investment on IT expenditures; the evolving expertise of GIS staff in areas such as system and database administration, web and other advanced programming languages; cooperative funding projects which resulted in hardware and software investment; and, the general recognition of the power that visualization and spatial analysis can provide.

Thus was the setting for the authors of this paper. We are two individuals working within the BIS GIS Section who were tasked to fuse and promote FDEP's ArcIMS development efforts by creating a secure, centrally maintained system for hosting ArcIMS sites and applications. The paper that follows contains a description of the system built, the FDEP Web Mapping System (FDEPWMS), and the lessons learned and challenges faced designing and constructing it.

One final introductory note to serve as disclaimer- the system described, the solutions presented, best suited and answered the specific needs of the GIS environment at FDEP. The same deployments and decisions will not work for all situations and organizations. The goals of this paper are to present our decisions and solutions within context and to hopefully aid any colleagues seeking information on a similar configuration or struggling with the challenges of designing an enterprise ArcIMS system.

## **2 FDEP WEB MAPPING SYSTEM (FDEPWMS)**

In September of 2002, we began the exercise to develop the FDEPWMS. Though we had complete responsibility and authority to design the system, our Section management defined the project's goals for us. These objectives framed our vision of the system and continue to drive its current maintenance, administration and development. The objectives are listed below in order of priority:

1. Security
2. Redundancy (Reliability)
3. Performance

Each objective and its realization in the system design are detailed in individual sections to follow. The design led us to a scalable, distributed array of five Wintel servers that would mitigate as many risks as possible while still satisfying the performance goal. The System consists of two web/application servers and two spatial servers (map servers) housed securely behind a firewall with only a Linux-based reverse proxy server available for public access in the De-militarized Zone (DMZ). A staging system that mirrors the production system evolved to test applications prior to deployment on the production array. The hardware and software platforms for the FDEPWMS were configured and optimized for their individual role in the system. Platform choices were also proven solutions that met Departmental IT standards ensuring support from our Network and Systems group and familiarity within our GIS development community.

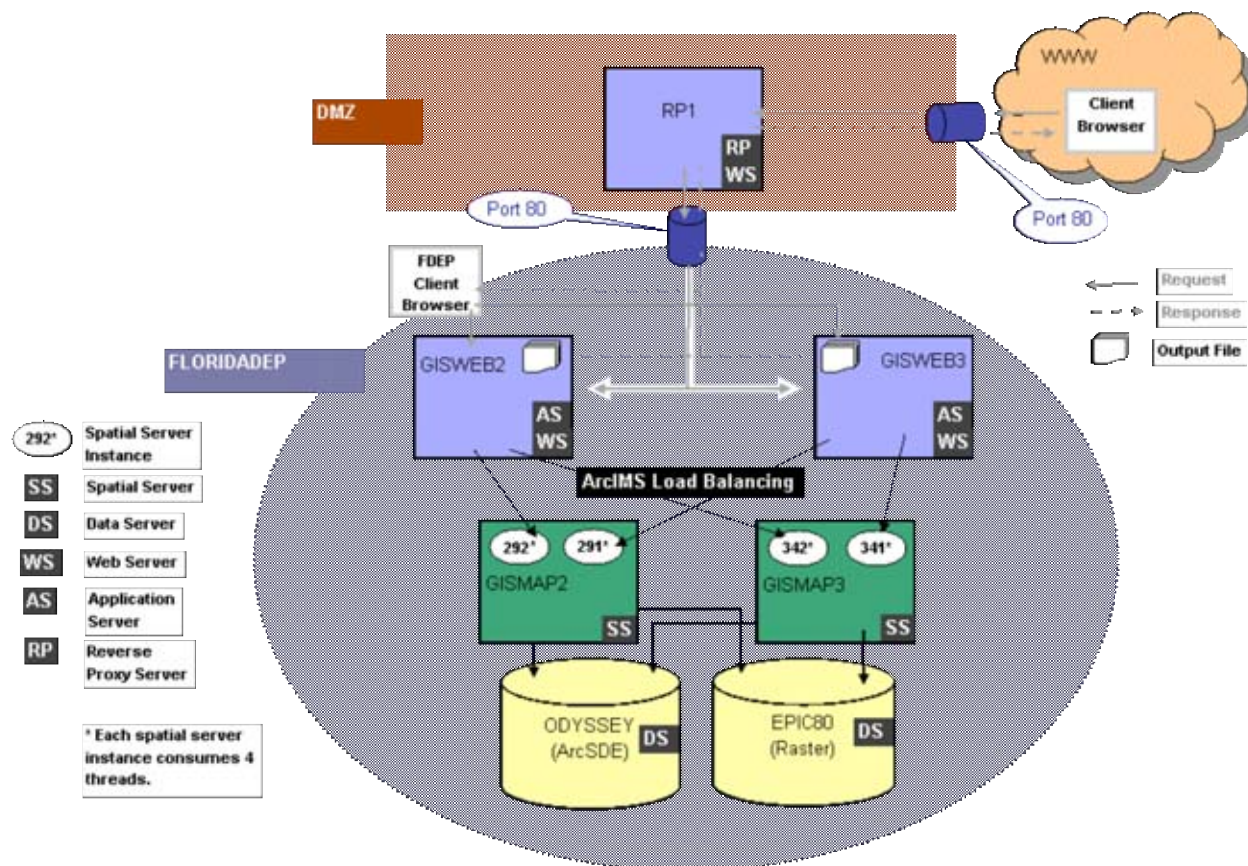


Figure 1: FDEPWMS Production System Configuration

With the exception of RAM, the two web/application servers and two spatial servers are configured with identical high-end Wintel hardware, sporting dual processors and SCSI RAID 0 disk arrays. Since the spatial servers perform the more resource intensive role in the distributed system, they are equipped with two gigabytes of RAM. The web/application servers have one gigabyte of RAM.

All four servers on the protected network are running Windows 2000 Server. The Web serving and mapping software running on the web/application servers are:

- Internet Information Services 5.0
- Tomcat-Jakarta 3.2.3
- ArcIMS 3.1 Application Server
- ArcIMS 3.1 Application Server Connectors
- ArcIMS 3.1 Manager
- Java Development Kit (JDK) 1.3.1\_02
- Java Runtime Environment (JRE) 1.3.1\_02

The spatial servers have the following mapping software installed:

- ArcIMS 3.1 Spatial Server (Monitor)
- Java Runtime Environment (JRE) 1.3.1\_02

Finally, the reverse proxy server is running Debian Linux OS and Squid Proxy Server. The box is configured with mid-range dual processors and one gigabyte of RAM.

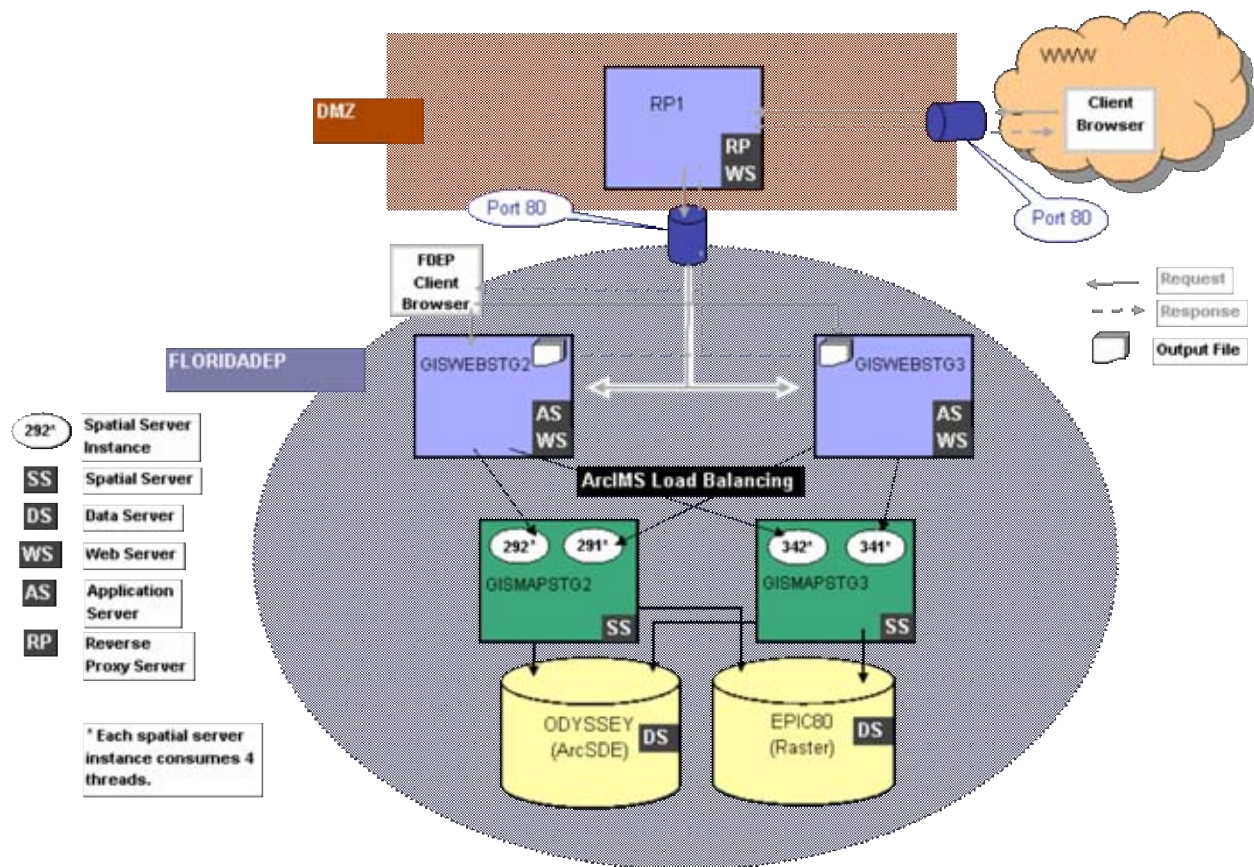


Figure 2: FDEPWMS Staging System Configuration

The staging system is configured identically using two web/application servers and two spatial servers. Though software and system configurations are duplicated on staging, there is a difference between the hardware used in the two systems. One half of the staging system consists of recycled GIS desktops. This variance, however, causes no overall impact. Also, since the two staging web/application servers are available through the reverse proxy server, outside communication can be thoroughly tested prior to moving applications to the production array.

## 2.1 Goal 1: Security

Our efforts to meet the Security objective can be summed up with the decision to implement a reverse proxy server for our ArcIMS configuration. Additionally we strove to limit our servers' vulnerabilities by first employing Industry best practices for securing all the servers, then applying an additional layer for the two web/application servers by using such tools as IIS Lockdown and URLScan.

The idea to use a reverse proxy server was first introduced to Chris in an exercise during an ESRI ArcIMS Administration class he attended in July 2002 (ESRI, July 2002). Intrigued by a configuration that mitigated the high risk of opening file-shares, an unavoidable consequence when opting for a distributed ArcIMS configuration, we began to work with our Network and Systems group to research its plausibility. Implementing a reverse proxy server would allow us to have only one box in the DMZ, with the rest of the boxes inside our FLORIDADEP network.

The reverse proxy server would function by tunneling HTTP requests from Internet users to the web/application servers behind the firewall using only port 80. Employing Linux as its OS offered yet another layer of security. After initial research and testing, the Network and Systems group offered their support. They also agreed to the administration and maintenance of that component of the system.

Our supplemental efforts to secure the system began with following recommendations outlined in Microsoft's *Windows 2000 Server Baseline Security Checklist*, including removing the default Administrator account, changing the Startup Type of several NT services, enabling auditing of object access in Local Security Settings and disabling or deleting any unnecessary accounts (Microsoft, 2001). Additionally, we followed the *IIS 5.0 Baseline Security Checklist*, which advised running the IIS Lockdown and URLScan utilities (Microsoft, 2001). As a precautionary note: some care should be taken not to lock down IIS so well that it prevents ArcIMS from working. Should you encounter any problems with IIS Lockdown, ESRI's Support Center provides documentation on this very subject (ESRI, Article ID: 23610).

Consequently, a tremendous secondary benefit from using a reverse proxy server is the ease of access for administrative and maintenance tasks for the ArcIMS servers, as they are within our internal domain. Developers also benefit from fewer headaches during trouble-shooting and error resolution as file transfer and ArcIMS administration tasks can be accomplished with little hassle. Another advantage is that this configuration allows our spatial servers to maintain faster network connections to our vector and raster data servers.

## **2.2 Goal 2: Redundancy (Reliability)**

Our second priority for the system was to build in reliability, which we could achieve through redundancy. Redundancy would help us reduce single points of failure by having duplicated machines in the case of hardware malfunction, while also providing load balancing to distribute server processing. If we could construct such a fault-tolerant system that minimized unscheduled downtime and bore commercial-grade qualities such as failover and load balancing, we might be capable of reaching that IT nirvana, otherwise known as, "high availability".

As the system developed, we began to understand that complete redundancy throughout the system was not going to be possible. Deadlines, sparse funding,



stretched staff resources and limitations in our internal network led us to appreciate that “high availability” for us is something more Zen-like in nature, a state you continually seek, but may never quite reach.

Recognizing where our system has limitations has helped us better plan for system improvements and compensate where possible. What we have attained is this: a level of service commitment that specifies system uptime goals within defined, scheduled hours; duplicate components where it is feasible; load balancing and failover configured where we could; and finally an acceptance that redundancy of the system will continue to evolve through funding and resource cycles.

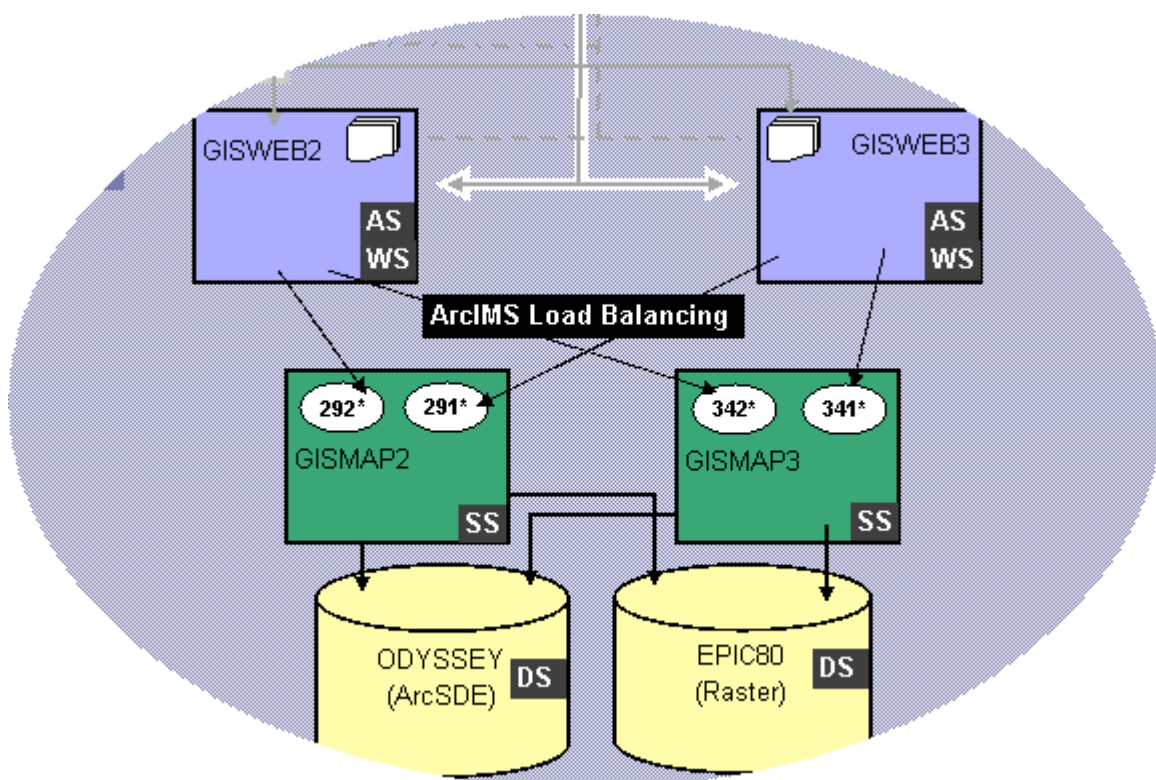


Figure 3: FDEPWMS Three-Tier Configuration Detail

Specifically, our efforts to approach “high availability” have led us to an ArcIMS design that features a three-tier, distributed system with separate platforms for the web/application server, spatial server and data server components. Given constraints mentioned earlier, hardware redundancy is addressed for the first two out of the three tiers. Dual web/application servers, configured identically with ArcIMS Application Server and ArcIMS Tasker, communicate with two spatial servers running ArcIMS Monitor. Both spatial servers request vector and raster-based data from a primary ArcSDE server and a primary raster server.

The machines that have the larger amount of RAM (2GB) were assigned to the spatial server role. Both spatial servers are configured with two separate instances of the ArcIMS Monitor executable- each instance is a NT service that supports communication

with one of the web/application servers. Installing two instances is necessary to enable ArcIMS load balancing which “distributes the spatial server processing load between the two map server platforms to avoid having requests back up on one server when extra processing resources are available on the other server” (ESRI, 2003). Also, spatial server failover allows the web/application servers to detect a failed spatial server and react accordingly by depending solely on the stable spatial server. The system remains operational during such an event and the server failure is transparent to clients with the exception of slower system performance.

Configuring the two spatial servers to run two instances of the Monitor NT service was one of the most difficult parts of this effort. It took considerable trial and error. We discovered that we needed a custom install utility for the additional Monitor process. The utility needed had to be a Java wrapper class that could install the Java classes required for running Monitor as an NT service. This is necessary because the Monitor executable installed by default will only support the running of one Monitor instance per machine (DeWeese, 2002). We were finally able to accomplish the feat after we found free software called JNT and an ESRI System Integrations Test Report that spelled out more clearly how to configure the spatial servers (DeWeese, 2001). ESRI now provides a service installer utility on request that will help in this effort (DeWeese, 2003).

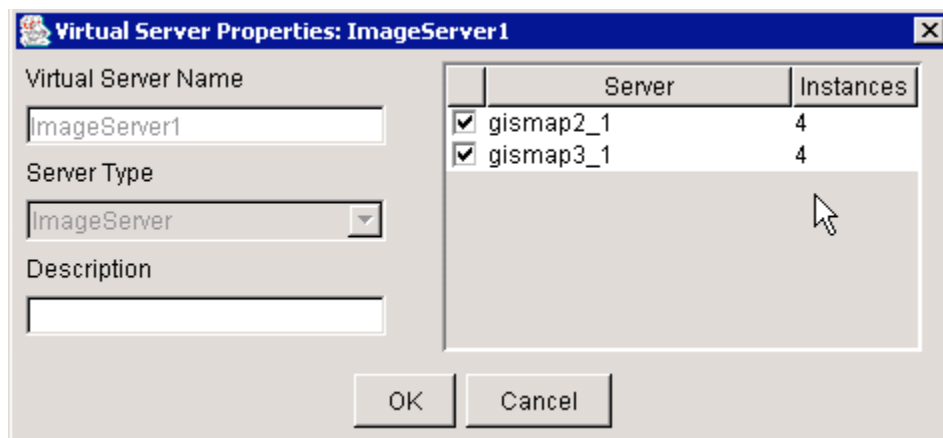


Figure 4: Additional Spatial Server Instances per Virtual ImageServer Detail

A final tuning and failover option relating to the spatial servers was to increase the number of spatial server instances assigned to each of our web/application server's Virtual ImageServers. The Virtual ImageServers on the web/application servers are part of the ArcIMS Application Server component and they handle requests for map services by passing them on to their registered spatial servers. By default each Virtual ImageServer had two instances per registered spatial server. Given our system's specifications, we increased this number to four, so that in the event of a web server failure our system would still be running at 100% capacity (DeWeese, February 2003).

In addition to hardware and software duplication, all paths and files needed for ArcIMS websites must be duplicated on the web/application servers and spatial servers. Map service files (AXLs) and websites must be duplicated on each web/application server

and any non-ArcSDE data must be copied to both spatial servers. This duplication is necessary so that an individual Map Service can be started as the same name on each web/application server and therefore is accessible from either web/application server.

An effort to load balance requests made to access websites on the FDEPWMS is made through server-side code on the FDEP ArcIMS Internet ([http://www.dep.state.fl.us/gis/portal\\_internet.asp](http://www.dep.state.fl.us/gis/portal_internet.asp)) and Intranet portal websites. Load balancing between the two web/application servers is simulated by randomly directing users to one or the other web/application servers when a hyperlink is chosen for a particular website. Additional efforts to increase the reliability of the system include using Industry standards such as dual power supplies and RAID 0 hard drive configurations on all the servers.

Finally, plans for including another reverse proxy server to add redundancy in the DMZ are already in the works. IP Load Balancing and ArcIMS Load Balancing which would further our efforts toward “high availability” will be discussed further in the Future Plans section of this paper.

### **2.3 Goal 3: Performance**

Performance and speed have always been the least important objectives in the development of the FDEPWMS. Understandably so, since we are not part of a private, commercial enterprise, but rather work with a State agency that funds the system with public, tax-payer dollars. For the most part, performance attributes have been the result of implementations made to support system redundancy. Primarily performance is improved by our adoption of a design that includes multiple spatial servers for each web/application server. Consistently written throughout ESRI's documentation is the recommendation that initial system capacity and performance improvements can be made easily by including additional spatial servers (ESRI, 2002).

To a lesser degree we have boosted our system's performance through a distributed system design and the ArcIMS spatial server load balancing efforts. This option routes requests from web/application servers to the most available spatial server. Also specifying more RAM for the spatial server hardware has helped with performance issues. Finally, efficient development practices followed by our development community help lift performance. Examples include: optimizing and tuning AXL tag references in map service files and requests and using ArcSDE layers that are generalized specifically for ArcIMS use, whenever possible.



### **3 LESSONS LEARNED**

The following section includes an overview of the key points we learned during our challenge to build the FDEPWMS. These suggestions are items that came up more than once in our post-mortem discussion on the FDEPWMS development project, or they are topics that have sprung up in more recent talks as the system has evolved. For the most part, these are lessons that we want to share because we wished we had known them before we started and they have often saved us time and headaches.

#### **3.1 Create Separate Systems for Development, Staging and Production**

“Never ever develop and test on your production system,” that statement comes from a slide in one of the ArcIMS Technical Workshop presentations at the ESRI User Conference and we have taken it to heart (Russo, 2002). We initially built the FDEPWMS as the production system and the staging system came later. The staging system has been one of the most important evolutions of our system. It is crucial for our developers and for us as administrators because it allows us to test applications in production conditions without risk. We have made a strident effort to completely replicate the production system’s configuration so our testing is as accurate as possible. The staging system has also been invaluable for testing security patches, OS updates and new ideas for administrative and monitoring tasks.

Also, because we stress an ArcIMS development cycle that includes primary development and testing on single, standalone systems, the staging system is the first opportunity to test any issues involving a distributed system. The primary objective, however, is to iron out problems related to the hosting of the site, not to further test functionality.



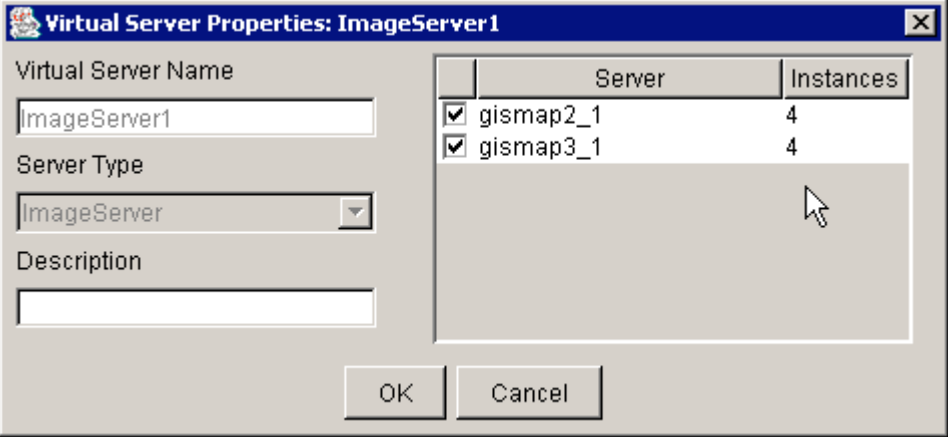
#### **3.2 Use Generalized Layers**

Building on knowledge we gained from another ArcIMS research effort in 2001, we highly suggest taking the effort to generalize reference data layers used in ArcIMS websites (Clay, 2001). Though the research pointed out that not all data benefits from generalization, usually improvements in map generation time are realized by reducing the number of features, vertices and/or attributes from the data layer. Generalized layers are not a panacea, however. Other tuning methods exist that help reduce map generation time, such as the efficient use of AXL code for scale-dependencies on layers or when labeling, for performing spatial queries and when choosing the optimal output image type (Russo, 2002).

### 3.3 If Applicable, Monitor ArcSDE Connections

ArcSDE connections are a limited resource in our environment. As mentioned before, we have over 350 internal GIS users who consume ArcSDE data in ArcGIS 8x, ARCINFO 7x, ArcView 3x and now through the FDEPWMS, public users consume connections, as well. Our section maintains the Department's centralized, enterprise spatial database server, which currently runs ArcSDE 8.1.2 and ORACLE 8.1.7.2.0. Since the FDEPWMS is a distributed system with multiple spatial servers and an increased number of spatial server instances, we have had to test, monitor and plan system changes that cause increases in the use of ArcSDE connections. We have developed a formula that helps us predict the number of ArcSDE connections that changes or additions to the system will make:

$$(a) \times (b) \times (c) \times (d) = \# \text{ of ArcSDE connections}$$

<b>a</b> = Number of ArcSDE users in AXL file	<WORKSPACES> <SDEWORKSPACE name="sde-1" server="XXX" instance="port:XXX" database="" user="XXX" encrypted="true" password="XXX"/> </WORKSPACES>																							
<b>b</b> = Number of SS Services	 ArcIMS_SS01      ArcIMS Monitor - uses GISWEB3  ArcIMS_SS02      ArcIMS Monitor - uses GISWEB2																							
<b>c</b> = Average number of instances Per SS Service																								
<b>d</b> = Number of Virtual Servers using ArcSDE (ImageServer, GeocodeServer...)	<table><thead><tr><th>Virtual Server Name</th><th>Description</th><th>Access Type</th></tr></thead><tbody><tr><td>FeatureServer1</td><td></td><td>PUBLIC</td></tr><tr><td>ExtractServer1</td><td></td><td>PUBLIC</td></tr><tr><td>ImageServerWRM1</td><td></td><td>PUBLIC</td></tr><tr><td>ImageServer1</td><td></td><td>PUBLIC</td></tr><tr><td>GeocodeServer1</td><td></td><td>PRIVATE</td></tr><tr><td>QueryServer1</td><td></td><td>PRIVATE</td></tr></tbody></table>			Virtual Server Name	Description	Access Type	FeatureServer1		PUBLIC	ExtractServer1		PUBLIC	ImageServerWRM1		PUBLIC	ImageServer1		PUBLIC	GeocodeServer1		PRIVATE	QueryServer1		PRIVATE
Virtual Server Name	Description	Access Type																						
FeatureServer1		PUBLIC																						
ExtractServer1		PUBLIC																						
ImageServerWRM1		PUBLIC																						
ImageServer1		PUBLIC																						
GeocodeServer1		PRIVATE																						
QueryServer1		PRIVATE																						

Currently our configuration is  $(1) \times (2) \times (2.5) \times (4) = 20$

**a** = 1 SDE user

**b** = 2 SS (spatial server) NT Services

**c** = 2.5 Average # of instances per SS service

**d** = 4 Virtual Servers using ArcSDE (ImageServer1, ImageServerWRM1, QueryServer1, ExtractServer)

Additional information is available as a FAQ from ESRI's Online Support Center (ESRI, Article ID: 21954).

### **3.4 Manage Remote Development**

The FDEPWMS is a hosting environment built to potentially accommodate ArcIMS websites from any group within the FDEP. Though managed by our section, BIS GIS, the system is a shared resource. Therefore, we had to devise a management scheme that would give our developers the freedom they required, while not jeopardizing the integrity of the overall system. The final outcome was to set up user groups and NT shares for the system, create guidelines for access, develop an ArcIMS Development Plan document that helps us estimate resource usage and begin a list serv just for the FDEP ArcIMS development community.

In order to effectively manage file access to our distributed development community we decided to make file-shares from the website and AXL directories on the web/application servers and from the data directories on the spatial servers. Each Division in the agency is provided a directory inside each file-share. To meet security guidelines, a given Division only has read and write access to files within their Division's directories. NT user groups were created for each Division. As developers received access to the system, their domain account was added to the Division user group. Using NT user groups greatly eased our user account management tasks.

Beyond file shares, we had to draft access procedures which outlined the responsibilities of system use and the steps needed to port sites from a single development system to the multiple servers that make up the staging system and production array. Before we grant access, however, we require that developers provide us a plan document that better positions us to predict the usage of enterprise resources, such as ArcSDE. This document also helps us determine the scheduling of system improvements and aids in the final determination of supported technology. In addition to other information, the FDEP ArcIMS Development Plan furnishes us with:

- site usage estimates;
- the ArcIMS virtual servers to be used;
- names of Departmental databases and ArcSDE layers accessed;
- server-side or client-side technology employed in the site; and,
- a copy of the working AXL and URL to their development site

The Development Plan also gives us an opportunity to oversee and enforce guidelines laid out in the access procedures, including using the appropriate ArcSDE username/password combination to conserve connection resources. Once the system administrator has approved the plan, it is then cleared by the ArcSDE database administrator. After that, appropriate permissions are set on the system and the developer is given clearance and direction on where and how to port his/her application to staging. After sufficient testing is complete on staging, authorization is provided to proceed with moving the application to the FDEPWMS production array.

Finally, we also created an internal list serv, ARCIMS-L, to promote communication within the development community.

### **3.5 Avoid Large Image Catalogs**

We have had to figure out a way to deal with large image catalogs since providing imagery as a backdrop to vector layers or using it as a verification layer is a very popular option for FDEP ArcIMS websites. Catalogs that reference the Department's collection of 1999 or 1994 Digital Ortho Quarter Quads (DOQs) can have almost 4000 records and are stored on a separate, dedicated server. Early on when working on our single systems, we discovered we had to increase the extendedTimeout property in the AppServer.properties file to successfully start a map service with imagery (Fleming, 2003). Since then, we have learned that when a map service is started with large image catalogs such as ours, the spatial server reads the header of every image file listed in the catalog. Due to our distributed system configuration and the additional number of spatial servers we maintain, it can take upwards of an hour and \_ to refresh our map services.

We do have several raster-based data layers in ArcSDE, such as several scales of Digital Raster Graphics (DRGs) for the state of Florida. We are also trying to obtain the funding for additional storage that would support loading the DOQs in ArcSDE, but that project has a long time horizon. In the near term we are developing tools that will dynamically add file-based raster data to our ArcIMS sites.

### **3.6 Monitor the System**

Monitoring becomes increasingly difficult as the system becomes more distributed and complex. We use a combination of tools and procedures to verify that our systems are up and communicating, but we know there is room to improve our efforts in this area. Software and system logs are also key instruments in tracking system health, maintenance activities and in trouble-shooting efforts. Several free tools that help with monitoring ArcIMS systems are available from the ArcScripts portion of ESRI's website.

### **3.7 Preventative Maintenance is Good Medicine**

As the FDEPWMS system has evolved, we have instituted several preventative measures to maximize its availability. We use several scheduled tasks that run batch files to help keep the system running consistently and smoothly. To help prevent memory leaks and orphan processes, we stop and start the IIS and Jakarta services daily. We also refresh the ArcIMS Application Server, Monitor and Tasker NT services daily on each system because communication to our ArcSDE server is interrupted during database refreshes. We also use ADMINCMD AXL tags to run ArcIMS Administration tasks that essentially refresh and save our ArcIMS Administrator catalogs every morning. Finally, we reboot the system weekly and maintain a system

log to note daily system checks, system problems and resolutions, and any installs or changes made to the system.

## **4 FUTURE PLANS**

At present our future plans are focused on migrating to ArcIMS 4.01. After we have successfully built staging and production 4.01 platforms, we will port our existing ArcIMS applications and sites to these systems. At that point, we will be able to focus on other system improvements.

Three of our most important improvements will help on the reliability front. First, we have already purchased another reverse proxy server to include in our system and reduce that single point of failure. Next, we are planning on testing IP load balancing to divide and distribute HTTP requests passed from our reverse proxy server to the web/application servers. Also, in the event of a system failure this improvement will help with forwarding all requests to the available server. We plan to use Cisco hardware for this effort and will work with our Systems and Network Group to implement and test its success. Finally, we will look at ArcIMS Application Server load balancing as a means to contribute yet a higher level of availability to the FDEPWMS. ArcIMS Application Server load balancing gives us an extra layer of reliability by transferring the load of a failed or over-burdened ArcIMS Application Server to an available ArcIMS Application server.

## **5 CONCLUSION**

In mid 2003, ArcIMS is used in almost a dozen FDEP websites and online applications to provide geographic information to the public and internal staff. Functionality ranges from simple map display and query, to highly, customized sites offering server-side solutions that allow interactive data update and verification tools. The development of web mapping applications and viewers and the conversion of existing GIS tools to a web-based platform has become the major focus of GIS development within the agency. The continued development of the FDEPWMS and the web mapping systems that will follow it are essential if FDEP expects to keep pace with the growth and demand for web-based GIS tools and information displays.



## **6 ACKNOWLEDGMENTS**

To a large part our efforts in the development of the FDEPWMS would not have been successful or even possible without the direct or indirect support of the following individuals:

Linc Clay

BIS GIS Section members: Johnnie Abron, Eric Brockwell, Deborah Parker and April Williford

John Willmott

David Schrader

Ellen McCarron

Richard Butgereit and the WRM GIS Staff

Jeff DeWeese

Caroline Staab

BIS Network and Systems Group

David Judd

Guy Browning

Jorge Lagos

Bud Cates

Jack Woodard

Keith Rowe

## REFERENCES

- Clay, L. and Davis, D. (2001). Generalizing Features to Improve ArcIMS Performance: A Practical Guide 2001. ESRI 2001 User Conference Proceedings.
- Dawson, Marlyn E. (1992). An Assessment of Geographic Information System User Needs Within the Florida Department of Environmental Regulation (October 1988 – January 1989).
- DeWeese, Jeff, and Ray Echevarria. (2001, December 6). ArcIMS 3.1 High Availability Configuration Testing Using Network Load Balancing. ESRI's Enterprise Technology Portal. Retrieved November 20, 2001, from [http://eslims.esri.com/miscfiles/ArcIMS\\_31\\_NLB\\_Configuration.pdf](http://eslims.esri.com/miscfiles/ArcIMS_31_NLB_Configuration.pdf).
- DeWeese, Jeff, and Ray Echevarria. (2003, April 15). ArcIMS 4.0.1 High Availability Configuration Testing Using Network Load Balancing. ESRI's Enterprise Technology Portal. Retrieved June 20, 2003, from [http://eslims.esri.com/miscfiles/ArcIMS\\_401\\_NLB\\_Configuration.pdf](http://eslims.esri.com/miscfiles/ArcIMS_401_NLB_Configuration.pdf).
- DeWeese, Jeff. (2003, February 24). Follow-up question for J DeWeese. E-mail to Dixie Davis.
- ESRI Support Center. (2002, November 25). Error: The URL http:// is invalid or ArcIMS is not running. Article ID: 23610. Retrieved June 20, 2003, from <http://support.esri.com/index.cfm?fa=knowledgebase.techArticles.articleShow&d=23610>.
- ESRI Support Center. (2002, June 18). FAQ: How are ArcSDE Connections Allocated when Serving Data Through an ArcIMS MapService? Article ID: 21954. ESRI FAQ document. Retrieved June 20, 2003, from <http://support.esri.com/index.cfm?fa=knowledgebase.techArticles.articleShow&d=21954>.
- ESRI. (2002). ArcIMS Administration Lectures: July 2002. Redlands, CA.: Author.
- Fleming, M. (2003). Problem adding IMS service containing large raster catalogs. Posting on the ESRI ArcIMS User Forum Web Site. Retrieved June 20, 2003, from <http://forums.esri.com/Thread.asp?c=64&f=778&t=80696&mc=20#msgid245195>.

Microsoft TechNet. (2003, June 20). IIS 5.0 Baseline Security Checklist. Retrieved June 20, 2003, from <http://www.microsoft.com/technet/security/tools/chklist/iis5cl.asp>.

Microsoft TechNet. (2001). Windows 2000 Server Baseline Security Checklist. Retrieved June 20, 2003, from <http://www.microsoft.com/technet/security/tools/chklist/w2ksvrcl.asp>.

Peters, Dave. (2003, February ). System Design Strategies. ESRI's Enterprise Technology Portal. Retrieved June 20, 2003, from <http://www.esri.com/library/whitepapers/pdfs/sysdesig.pdf>.

Rosso, A. (2002). Growing an Internet Mapping System Using ArcIMS. ESRI 2002 User Conference Proceedings.

## **AUTHORS**

Dixie Davis  
GIS Contractor  
The GIS Shop, INC.  
PO Box 14124  
Tallahassee, FL 32317-4124  
dixie@gisshop.com

Chris Judd  
GIS Systems Project Analyst  
Florida Department of Environmental Protection  
Bureau of Information Systems  
2600 Blair Stone Road, MS 6520  
Tallahassee, FL 32399-2400  
chris.judd@dep.state.fl.us