

GeoContext: Putting Geospatial Data in Context with Web Documents

Nate Ryan, Kim Berger, Jon Strother

Abstract

GeoContext is a framework to display GEOINT data in a format that is easily understood and interpreted. The persistent table of contents in the left-hand column allows for the integration of both geospatial and non-geospatial data in a unified document that is presented in a meaningful context and in a visually appealing design. Geospatial data is integrated into the document using an ArcIMS® customization. Implementation of GeoContext -- accessible over the web at various security levels -- allows any organization to develop a country, regional or issue-based intelligence document of its area of interest addressing specific topics of concern. The included Customization Guide ensures rapid implementation while maintaining a consistent look and feel for all instantiations. GeoContext was created by NGA Innovision Directorate personnel within the St. Louis Geospatial Intelligence Advancement Testbed (GIAT) lab.

Background and Design Concept

A branch within the Production (P) Directorate of the National Geospatial-Intelligence Agency (NGA) was tasked to update an existing, but outdated, Atlas of Military Power. Recognizing that a hardcopy atlas becomes obsolete as soon as it is printed, they approached GIAT for assistance in finding a way to produce a softcopy atlas which, in theory, could be kept current with the most up to date data available. The requirements set forth for the project included:

- the ability for NGA P Directorate personnel to do data updates,
- the incorporation of both geospatial and non-geospatial content,
- a persistent Table of Contents so users are always aware of where they are in the document, and
- the aesthetics of a hardcopy atlas – i.e. that it look and feel like an integrated document.

Initially, two possible tools were discussed: Macromedia Flash™ and ESRI® ArcIMS®. Pros and cons of both were weighed. Flash was judged too difficult to rapidly update, which would result in the same 'obsolete upon release' problem as hardcopy. ArcIMS® was thought to be inappropriate in that it does not allow for non-geospatial content or support a persistent Table of Contents (TOC) for multiple page documents because it takes up the entire browser window. Upon research, GIAT found that ArcIMS® could be wrapped into a frame within a browser. This wrapping of ArcIMS® in a frame makes it possible to design an ArcIMS® project that can exist within a unified GEOINT document with a persistent TOC while also accommodating non-geospatial pages.

Over time, GIAT came to realize that the atlas design could be used for other types of GEOINT documents. The integration of non-map and map-based pages

in a unified document, driven by a persistent TOC, makes GeoContext a powerful delivery engine for nearly any GEOINT document. With it, an office can develop a country, regional or issue-based intelligence document of their area of interest (AOI), addressing specific topics of concern.

Overview

GeoContext differs from most web-based geointelligence sites in that it is a combination of PHP-generated HTML pages and ArcIMS® pages in one unified document. The PHP-generated portion can contain any standard web-based content, including static HTML, dynamic HTML, JavaScript®, graphics, PDF, and Flash animations.

The ArcIMS® portion differs from most ArcIMS® implementations, in that it only occupies part of the browser, and has been considerably modified from the out-of-the-box ArcIMS® HTML Viewer. Forcing ArcIMS® to occupy only a portion of the browser allows for a persistent TOC in the left frame, thus creating a unified document. HTML Viewer modifications include incorporation of user requested features, bug fixes, and enhanced navigation.

Navigation throughout a GeoContext document is handled by the TOC, found on the left side of the web page. Upon opening the document, the TOC displays the document's main topics. Initially, the TOC is unexpanded. TOC headings that contain content appear in blue and are underlined. Some of these items may have subtopics. Clicking on a topic which has a subtopic will cause that portion of the TOC to expand, exposing the available subtopics (Figure 1). As an aid to navigation, the current location within the document is always highlighted in light grey, as seen in both examples in Figure 1.

Table of Contents	Table of Contents
Home	Home
Executive Summary	Executive Summary
Military Forces	Military Forces
Transportation	Missiles
Comprehensive Map	Logistics
References/Bibliography	Transportation
Other Related Geospatial	Comprehensive Map
Intelligence	References/Bibliography
Products/Websites	Other Related Geospatial
Appendices	Intelligence
	Products/Websites
	Appendices
Help	
	Help

Figure 1: An unexpanded TOC (left), and an expanded TOC (right).

The right-hand portion of the web page can display either HTML or ArcIMS® content. Non-ArcIMS® enabled pages use a two column table to format the content, one column to hold the TOC and the other to hold the HTML content. ArcIMS®-enabled pages use a combination of framesets and frames to format the page content. Although the ArcIMS® and nonArcIMS®-enabled pages handle the TOC differently (frames vs. 2-column table), the difference is nearly transparent to the end-user. It becomes noticeable only when a TOC is longer than the display area. This dual design was driven by internal NGA web standards and was, therefore, unavoidable.

Technical Development

Software Selection Rationale

HTML Viewer: There are three viewer options within ArcIMS®: the HTML Viewer, the Java Viewer, and the Custom Java Viewer. These viewers are the interface choices for ArcIMS® pages created through the ArcIMS® Designer “wizards”. We selected the Custom Java Viewer initially due to its rich set of available GIS tools and the fact that it handles the legend display more elegantly than does the HTML Viewer. The Custom Java Viewer combines the legend and layer list as is done in all other ESRI® software. The HTML Viewer offers a toggle between a layer list and a legend. There are many technical issues associated with the Java Viewer, however, which caused us to abandon it. Chiefly, they are the lack of support for the Java Viewer to fix known bugs, and the fact that the application requires clients to use a specific JRE which we deemed unlikely at best. GeoContext ultimately used a customized version of the HTML Viewer.

ArcMap® Server: There are three map servers available for use with ArcIMS®. They are Image Server, Feature Server, and ArcMap® Server (AMS)¹. ArcMap® Server serves up the .mxd generated in ArcMap® through ArcIMS® via an ArcMap® Image Service.² GeoContext uses the ArcMap® Server. Several factors were considered in choosing to use the ArcMap® Server, with speed, ease of use, and software compatibility being foremost.

It has been documented that the ArcMap® Image Server takes about a 10% hit in speed for page rendering as compared to Image Server.³ Regardless of this fact, we chose to use ArcMap® Server because of its ease of use and the fact that extensions to ArcMap® can be used when serving up ArcIMS® content. One of the goals of this project was to develop a framework that Geospatial Analysts (GAs) would find easy to use, both for initial page generation and for updates. ArcMap® is a very familiar application to many of the GAs within NGA, which makes ArcIMS® map generation via AMS fairly intuitive. Using AMS in creating the ArcIMS® pages circumvented the need for GAs to learn and work with Arc Extensible Markup Language (AXL).

¹ Getting Started with ArcIMS, p. 5, ESRI, c 2004.

² Ibid, p. 11.

³ ESRI Systems Integration Technical Brief: ArcIMS Configuration Performance Factors, p. 4, ESRI, Nov. 12, 2005.

Military OverLay Editor (MOLE®): Software compatibility with MOLE® was also a major factor in choosing AMS. One of the stated requirements for the initial project was the incorporation of Military Standard 2525B symbology. To accomplish this, we decided to use MOLE®, an ArcMap® extension that builds the 2525B symbology on-the-fly based on feature attributes. This ArcMap® extension is functional within ArcIMS® only if an ArcMap® Image Service is being used.

Data Delivery Extension (DDE): The standard ArcIMS® data download tool is incompatible with ArcMap® Server. The Data Delivery Engine (DDE) is an ArcIMS® extension that offers data download in a variety of data formats and coordinate systems. It was created by SAFE Software in conjunction with ESRI®. Because DDE works with any kind of map service and offers robust functionality, it was our choice to accommodate data download. Without incorporating DDE, ArcIMS® pages served up via AMS would have no download capability.

PHP: Because of some issues with JavaScript® and Internet Explorer®, we changed the dynamic program language of the TOC system to PHP. The specific problems encountered are discussed later in the Technical Challenges and Solutions section under “Overcoming the TOC Showstopper”.

Directory and File Schema

Below is an outline of the directory and file schema for GeoContext. Note that, although the ArcIMS® webpage directory example is shown below under the TOC Sub-Sub-Topic directory, they can reside at any directory level under the document root directory. Also, NGA Gateway rules require that *all directory and filenames be lowercase* for non-ArcIMS® webpage directories.

Document Root Directory/

menu.inc, menu.php, menu.txt, *.php, *.html, <theme>tabs.js file(s) (optional)

help_images/

*.jpg, *.gif

images/

*.jpg, *.gif

TOC Topic Directory/ (e.g. pol_mil_info)

*.php, <theme>tabs.js file(s) (optional)

images/

*.jpg, *.gif, etc.

TOC Sub-Topic Directory/ (e.g. milforces)

*.php, <theme>tabs.js file(s) (optional)

images/

*.jpg, *.gif, etc.

TOC Sub-Sub-Topic Directory/ (e.g. logistics)

*.php, <theme>tabs.js file(s) (optional)

images/

*.jpg, *.gif, etc.

ArcIMS® Webpage Directory/Directories/ (e.g. ims_logistics)

*.htm, *.html, *.js

javascript/

*.js

images/

*.gif, *.jpg

symbols/

*.gif (<your_layer_name>.gif)

Both .htm and .html files can be found in the ArcIMS® webpage directory. Files ending in .html are files that were created by NGA's GIAT-St. Louis. Files ending in .htm are files originally created by ESRI®. We have edited several of ESRI's .htm files in our customization.

GeoContext Data Management Issues

This section discusses several data management issues which drive the ArcIMS® portion of GeoContext. These include layer naming requirements related to MOLE®, DDE downloads, and label placement, as well as feature attribution requirements related to hot-linking.

Layer naming requirements for MOLE®: Because GeoContext uses ArcIMS® ArcMap® Server, all that is required to drive the ArcIMS® map and legend is a properly structured .mxd file, using a combination of simple layers, group layers and MOLE® layers. Because AXL does not distinguish between different types of non-feature layers, we implemented a naming convention so that GeoContext could programmatically distinguish differing layer types in order to populate the GeoContext ArcIMS® legend properly.

Simple layers include raster and feature layers. Both of these layer types are viewable within the map. Raster and feature layers have no special naming requirements.

Group layers are collections of related layers. MOLE® uses group layers by default. MOLE® group layers include one feature layer and one MOLE® graphic layer. To indicate to the GeoContext ArcIMS® code that a group layer is a MOLE®-related group layer, the ArcMap® layer is named with "MGroup" as the last element of the name as follows:

```
<layer name> MGroup.
```

A parent group layer is a collection of related layers under a "parent" category heading. GeoContext does not support nested parent layers. Parent group layers can be set as either open (child layers exposed in the ArcIMS® legend upon initial load) or closed (child layers not shown in the ArcIMS® legend upon initial load). GeoContext uses the following naming system to control how parent group layers are displayed (opened or closed):

```
<layer name> Parent (#) <status>.
```

where <layer name> is the name of the layer as it will appear in the GeoContext ArcIMS® legend, "Parent" is a literal, a number in parentheses indicating the number of feature and MOLE® graphic layers contained in the parent group layer, and ending with a literal "Open" or "Closed". For example, if you had a parent group layer called Transportation containing one MOLE® group layer and two feature layers, the number in parentheses in the parent group name would be 4. This is because the MOLE® group layer contains one feature layer and one

MOLE[®] layer. If you want all the child layers within the parent group layer to be exposed upon initial load, the status would be “Open”.

A MOLE[®] layer is a graphic layer contained in a group with its corresponding feature layer. Graphic layers are for display only and cannot be queried as feature layers can. In addition, graphic layers are not symbolized in the ArcMap[®] legend nor in the out-of-the-box ArcIMS[®] HTML Viewer legend. To be displayed properly in the GeoContext legend, MOLE[®] layers must be named as follows within ArcMap[®]:

<layer name> (MOLE).

Figure 2, below, illustrates the relationship between the ArcMap[®] layer naming and grouping conventions and the resultant ArcIMS[®] legend display.

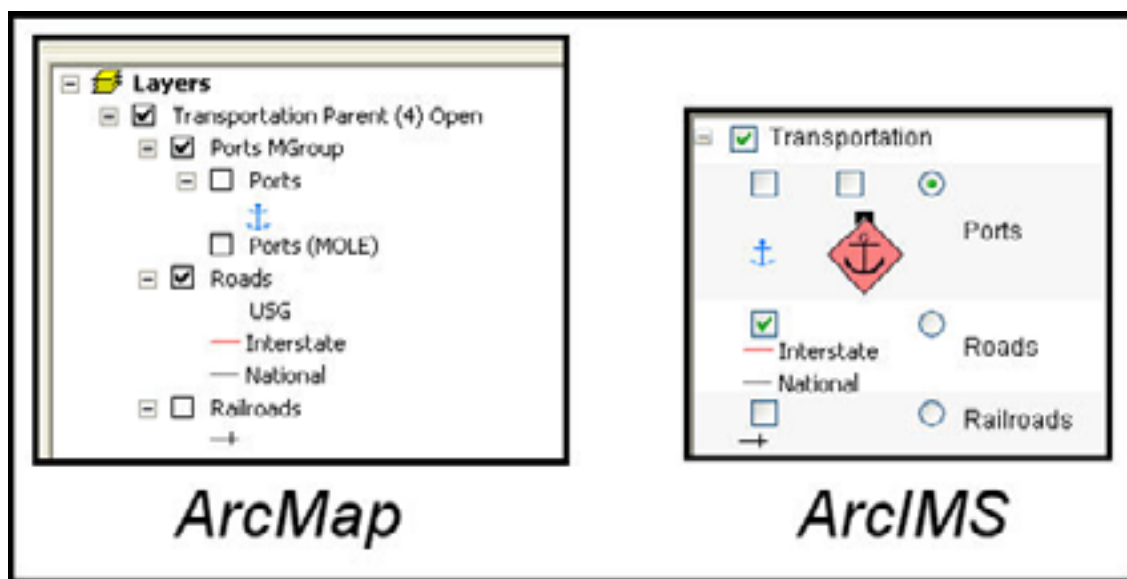


Figure 2: Relationship of ArcMap[®] Layer Naming Convention to ArcIMS[®] Legend Display. Layers can be simple layers (e.g. Roads) composed of features only, or group layers (e.g. Ports) composed of a feature layer and a MOLE[®] graphic layer. By following ArcMap[®] layer naming conventions, the proper names and symbols will be generated in the ArcIMS[®] legend display.

Hidden layers and DDE: In some cases, several layers point to one data source. For example, you may have warehouse, depot, and vehicle equipment layers all pointing to one logistics source (shapefile, geodatabase feature class, etc.). Even though these layers are for subsets of the logistics data source, DDE will download the entire data source file. This can be confusing to users as they may think they are only downloading a subset (e.g. warehouses). To avoid this confusion, add a layer in the ArcMap[®] MXD file that will be hidden in the ArcIMS[®] page. This new layer (e.g. “All Logistics”) will replace all subset layers from the same source in the DDE download list so that end-users get what they are expecting in the download.

Simplified polygons for label placement: Download times for ArcIMS[®] pages can be greatly enhanced by using labels wisely. It takes longer to place a label in a

detailed polygon than in a simple polygon. If your data has very detailed polygons that need labels (e.g. countries, states, provinces, or military regions), you can create grossly generalized copies of the polygon data to use for labeling instead. In the ArcMap[®] MXD file, simply make the simplified polygon layer have no fill and no outline so only the labels display. By then hiding the generalized “label” layers in the ArcIMS[®] legend and linking their display to the detailed layers, display time is greatly enhanced. Instructions on how to link the two layers for ArcIMS[®] display purposes is discussed in the GeoContext Customization Guide.

Technical Challenges and Solutions

Wrapping ArcIMS[®] into a Portion of the Browser Window: An ArcIMS[®] page typically takes up the entire browser window. In order to create a single unified document with a persistent TOC, we had to find a way to make ArcIMS[®] share the browser with other content. This is done by creating additional frames within the viewer.htm file to contain items such as the document TOC and Agency banners.

Simplifying the TOC Generation: Publishers of GeoContext documents are production personnel, and therefore are not expected to know PHP. They must, however, have a way to create their own TOC content. To avoid GeoContext users being required to edit PHP, we devised a method which uses a simple text file, menu.txt, for TOC generation. PHP reads this text file as data and uses it to automatically generate the TOC. Each uncommented line of the menu.txt file contains two double-colon (::) delimited fields. The first field contains the text that will appear in the document’s TOC. The second field is the absolute path to the PHP or HTML file controlling what appears to the right of the TOC in the document. If no second field is present, GeoContext creates a non-linked entry in the TOC that can be used as a heading. Additionally, tabs are important in the menu.txt file as they are used by GeoContext to determine position and indentation in the TOC.

The TOC shown in Figure 1 was generated with the following menu.txt file:

```

#text file to build Table of Contents
#   Use tabs at the beginning of the line to set hierarchy: zero tabs
for highest level entry - no more that 2 tabs allowed (3
hierarchies).
#   There are then 2 values separated by TWO colons (::)
#       The first value is the text that appears in the GeoContext
document's table of contents.
#       The second value is the href link (using the full URL) to
which this TOC value refers.
#   To create a heading with no link, simply enter the text for the
heading without adding the :: and the href link.
#
#
#   The pound sign (#) is used for comment lines
#

Home::/Website/geocontext/cover.php
Executive Summary::/Website/geocontext/execsum.php
Military Forces::/Website/geocontext/milforces/milforces.php
    Missiles::/Website/geocontext/milforces/missile1.php
    Logistics::/Website/geocontext/milforces/logistics1.php
Transportation::/Website/geocontext/trans/trans.php
Comprehensive Map::/Website/geocontext/compmap/compmap.php
References/Bibliography::/Website/geocontext/ref/ref.php
Other Related Geospatial Intelligence
Products/Websites::/Website/geocontext/other/other.php
Appendices::/Website/geocontext/appendix/appendix.php

Help::/Website/geocontext/help.php

```

Integrating Two Page Types: NGA has a firm requirement to avoid frames in web pages. This requirement is waived for ArcIMS® pages because ArcIMS® is itself a frames-based application. However, no waiver will be issued for the non-ArcIMS® pages of a GeoContext document. Therefore, GeoContext implements two page types: frames-based for ArcIMS®-enabled pages, and tables-based for nonArcIMS®-enabled pages. This does cause a slight difference in how the two page types handle, if the TOC is long enough to require vertical scrolling.

Supporting Two Levels of Toolbar Functionality: There are many tools available in ArcIMS®. However, we were advised that many potential end-users are overwhelmed by too many tools on a toolbar. To avoid intimidating some users, while not limiting others, we devised a toolbar toggle methodology. When an ArcIMS® page initially loads, the user is presented with “standard tools” (an Overview Map Toggle, Zoom In, Zoom Out, Pan, and Identify) and an Advanced Tools toggle button. Selecting the toggle button will load the “advanced tools” which include the four standard tools and a host of others. These include Zoom to Full Extent, Zoom to Active Layer, Back to Last Extent, HyperLink, Query, Find, Measure, Set Units, Display Coordinates, Buffer, Select by Rectangle, Clear Selection, Print, and Export Map Using DDE.

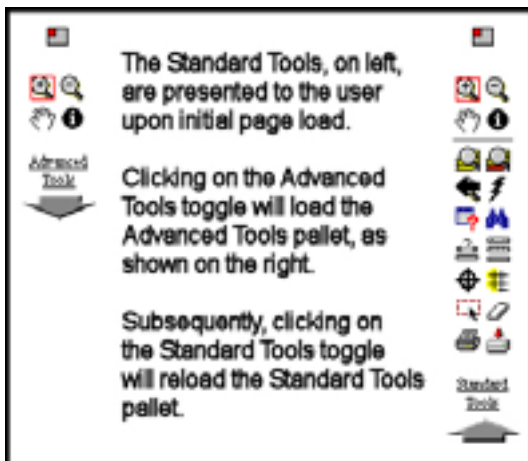
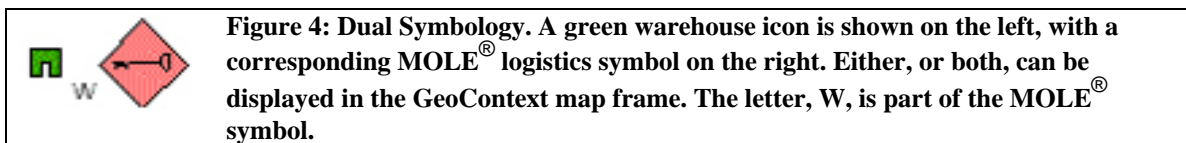


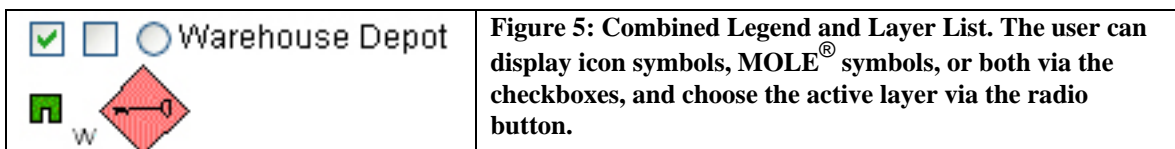
Figure 3: The toolbar toggles.

Accommodating Dual symbology: Our users wanted both icons and Military Standard 2525B symbology. To accomplish this, we developed a method for displaying both types of symbols in the GeoContext ArcIMS® legend. MOLE® generates 2525B symbology, but does not display these symbols in the ArcMap® legend or, by extension, in the ArcIMS® legend. It is, therefore, necessary to capture MOLE® symbols directly from the ArcMap® map canvas via screen shots, and then save each symbol as an individual gif.



We discovered that many of the feature categories displayed on the map do not have unique 2525B symbols. In Logistics, for example, there are no separate symbols for general logistics vs. naval logistics. We used the text label feature of MOLE® to make this distinction. This can be seen in Figure 2 above, where the W next to the MOLE® graphic indicates the type of logistics in this feature class – in this case Warehouse Depot.

Combining the legend/layer list: One of the advantages of the Custom Java Viewer over the HTML Viewer was the combined legend and layer list. When moving from the Custom Java Viewer to the HTML viewer we were unwilling to lose that functionality, so we devised a method to combine the two.



To accomplish this, we completely rewrote the legend frame. GeoContext includes the symbol images below the corresponding checkboxes. The images need to be captured from the ArcMap[®] display, saved into a specific directory, and named the same as the layer name. For example, if your layer name is Warehouses, you would capture the symbol for a warehouse from your ArcMap[®] session, save it to the <IMS page dir>/images/symbols directory as "Warehouses.gif". If there were a warehouse MOLE[®] symbol, you would save that image as "Warehouses_mole.gif" in the same directory. GeoContext, through JavaScript[®], looks for these images to use in building the legend automatically.

Speeding up label placement: Studies show that users are willing to wait from 5 to around 30 seconds for a page to load, depending on page design. For graphic pages, such as a map, the acceptable wait time is towards the lower end of this range⁴. We found that several layers in the map display were slow to load. Investigation determined that label placement for polygon features seemed to be the cause. To decrease load time, we created a second, hidden, polygon layer with grossly simplified polygons which is used only to place labels. We then programmatically link the detailed (unlabeled) and simplified (labeled) layers – displaying features from one, and labels from the other. This topic was discussed earlier in the GeoContext Data Management Issues section and is discussed in further detail in the customization guide.

Incorporating MOLE[®]: We discovered a bug when using MOLE[®] in conjunction with ArcIMS[®]. MOLE[®] allows for leadering and stacking as ways to minimize clutter on a map. These features work consistently in ArcMap[®], but do not carry over well to ArcIMS[®]. In ArcIMS[®] 9.1 MOLE[®] leaders do not group correctly, and stacking gives unpredictable results. Leadering and stacking should be avoided when constructing an MXD intended to drive ArcIMS[®] content.

Enhancing the Print Tool: The standard print tool in ArcIMS[®] tends to print map components in a serial manner. We wanted the layout of the printed map to fit the page better. In addition, the standard print tool will incorporate MOLE[®] (2525B) symbols in the map, but not in the printed legend. The print tool has been customized in GeoContext to handle the layout issues and to incorporate both MOLE[®] and dual symbology in the legend.

Overcoming the TOC Showstopper: We discovered there was a bug in how some versions of Internet Explorer[®] handled dynamic popup windows. Each time a popup window came up, the TOC became locked making it impossible to navigate further through the document. In order to solve this, we changed to the PHP language to drive the persistent table of contents and some other dynamic aspects of GeoContext.

⁴ Bailey, Dr. Bob. May 2001. "How Long Should Users Have to Wait?", in *Web Usability.com*. Retrieved 27 April, 2006 from www.webusability.com/article_how_long_should_users_wait_5_2001.htm.

Incorporating the Ability to Download Data: As stated earlier, the standard ArcIMS® data download tool is incompatible with ArcMap® Server. Therefore, we implemented DDE to manage data download. While installing and configuring DDE, we discovered several bugs. These included problems with feature classes that have an underscore in the name, rounding errors that caused the bounding box to fall outside the maximum or minimum extent, and the selected set option not working for Personal GeoDatabase layers. We worked closely with ESRI® Technical Support to solve or find workarounds for these problems.

In addition to finding workarounds for the bugs mentioned above, we developed methods to handle multiple layers from the same source, as described in the GeoContext Data Management Issues section. If multiple layers point to the same data source, the subset layers will need to be marked “no_download”, while a hidden layer pointing to the same source (“All <theme>”) will need to be marked as downloadable.

Including Classification Caveats: ArcIMS® is built using frames within the browser window. Consequently, certain security issues exist. Every frame within the browser has a discrete source file that can be viewed and/or printed out separately from the other frames. Because of this, each frame must have its own classification caveats for that individual frame. There is also the banner frame that holds the overall classification which must be derived from the highest classification combination of all the other frames on the page. Using JavaScript® and variables with “TEMPLATE” markings in the source code, these issues were resolved.

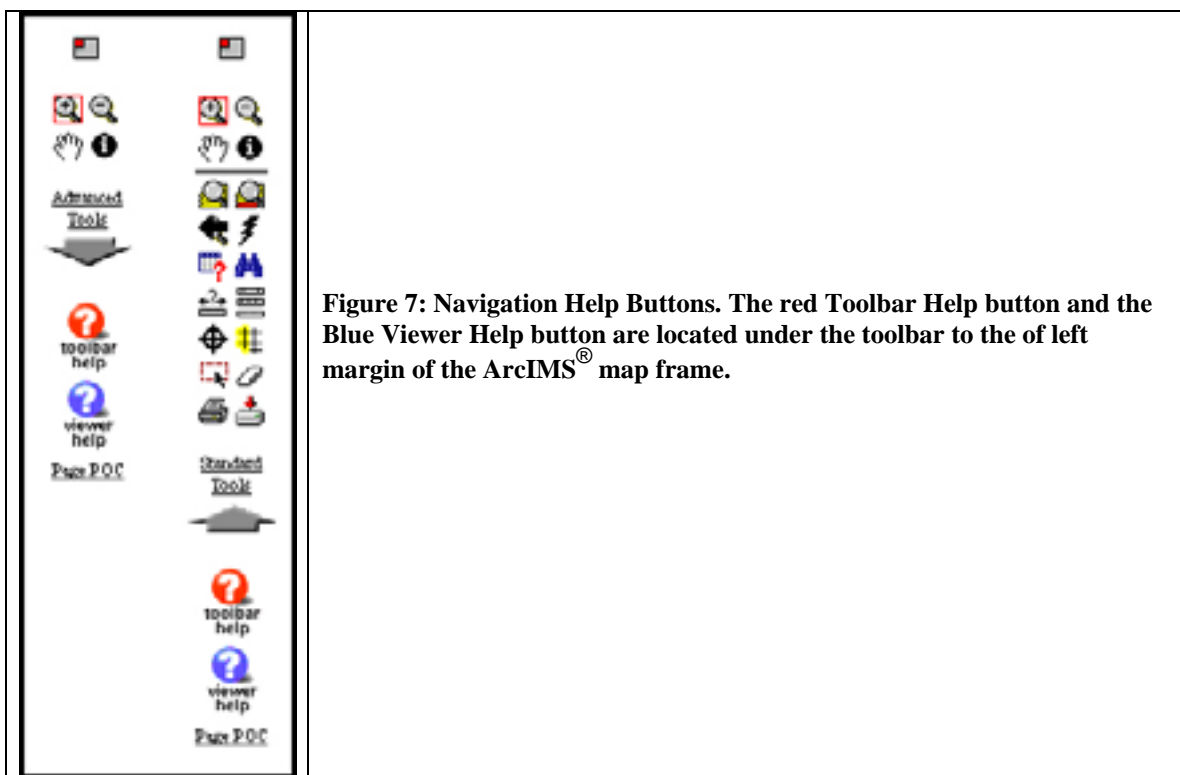
Handling Different Layer Types: AXL does not distinguish between different types of non-feature layers. In the GeoContext ArcIMS® legend, we wanted the ability to put the feature and MOLE® graphic layers side-by-side and also allow layers to be held within a parent group layer. In order to do this, we needed the ability to distinguish between layer types. Establishing ArcMap® layer naming conventions was key in solving this problem. See the Data Management Issues section under “Layer naming requirements for MOLE®” for details on the ArcMap® naming conventions.

Overcoming the Netscape® Loading Problem: In Netscape® 7 and other Mozilla®-based browsers, an ArcIMS® page never finishes loading and shows a message in the status bar that the data is still “transferring” even though the page has apparently completed. Coordinate information that is normally displayed in the status bar while moving the mouse over the map area, therefore, cannot be displayed. We noticed that clicking on a tool in the toolbar caused the “loading” to stop and the status bar to display the coordinates as it should. To fix this, we programmatically added a “ghost” tool (not shown in the toolbar) that is automatically “clicked” when the map finishes loading. This tool does not execute any commands, but it fixes the problem and is transparent to the user.

Help

GeoContext has various help files available, depending on context. There is a general Document Help, available in the persistent TOC, intended to provide a high level overview of how to use the current GeoContext document. This usually includes a section on document navigation and links to the two other help pages: Toolbar Help and Viewer Help.

Toolbar and Viewer Help, as well as subject matter expert contact links, are also always available from an ArcIMS[®]-enabled page, as seen in figure 7.



The Page POC link, located just below the Viewer Help button, will display contact information for the subject matter experts conversant on the topic of interest in the current map display.

Toolbar Help is driven by a one-row table of linked images. When the user clicks on a given tool image, they are driven to a table row that shows the tool icon, gives the tool name, and briefly describes what the tool is and how to use it.



Figure 8: A partial screen shot of the Toolbar Help page.

Viewer Help is driven by an image map of the overall viewer with clickable labels for each part of the viewer (Figure 9, below). Clicking on a label drives the user to a table entry that will display the name of that portion of the viewer along with a brief description of what it is and how it works. We borrowed the toolbar⁵ and viewer⁶ help design concepts from NOAA.

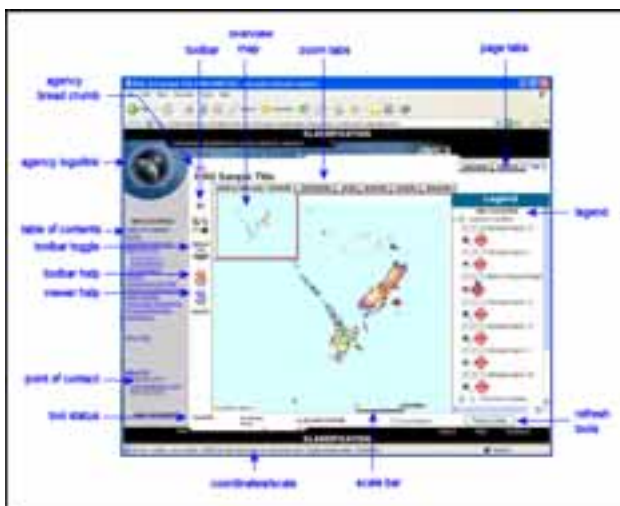


Figure 9: The Viewer Help image map.

⁵ NOAA. June, 2005. "Tool Bar Help" in *NGDC Interactive Map Services*. Found at <http://www.ngdc.noaa.gov/maps/arcimstoolbarhelp.html>.

⁶ NOAA. June, 2005. "ArcIMS Help" in *NGDC Interactive Map Services*. Found at <http://www.ngdc.noaa.gov/maps/arcimslayershelp.html>.

Customization guide

The GeoContext distribution includes a Customization Guide. The purpose of the guide is to explain how to prepare data for content and how to modify current template files to meet the needs of any office or organization. It demonstrates how to build web-based GEOINT documents while remaining consistent in design and maintenance over all implementations. Towards this end, the guide contains specific step-by-step “TEMPLATE” references where items may be modified by the document producers. Some of these modifications are required while others are optional. TEMPLATE flags use the following format:

```
<!--TEMPLATE: set the point of contact for the entire atlas -->  
    Generic content that needs to be changed.  
<!--TEMPLATE -->
```

By locating these flags within the code and making the appropriate changes to the portion of the code between the opening and closing flags, producers will be able to easily create a custom GeoContext document that remains true to the GeoContext design, and is consistent across organizational lines.

Demo

The accompanying demonstration was created to illustrate the functionality of GeoContext as a method to deliver an integrated intelligence document. All of the data and situations are notional to ensure availability to the broadest possible audience. It examines the issue of disputed off-shore oil fields between two fictional countries located in the Indian Ocean. The data and associated GeoContext document are included as part of the distribution and can be downloaded from [www.nga.mil/\[URL-TBD\]](http://www.nga.mil/[URL-TBD]).

References

ArcIMS Data Delivery Extension (DDE) Reference Manual. Safe Software, 2004.

Cardella, David, with Tim Bricker, Martin Flemming, and Rex Hanson.
Using Military Overlay Editor for ArcGIS. Redlands, CA: ESRI, 2004.

Customizing ArcIMS using HTML and JavaScript. Redlands, CA: ESRI
Educational Services, 2002.

Getting Started With ArcIMS. Redlands, CA: ESRI, 2004.

Lerdorf, Rasmus, and Kevin Tatroe. Programming PHP. Sebastopol, CA: O'Reilly
& Associates, 2002.

NOAA Satellite and Information Service, NGDC Interactive Map Services.
“Tool Bar Help” and “ArcIMS® Help.” June 2005.
URL: <http://www.ngdc.noaa.gov/maps/>

Author Information

Nathan Ryan, Jon Strother, Kimberley Berger
Project Scientists, NGA
3838 Vogel Road
Arnold, MO 63010
314-263-4924
nathan.d.ryan@nga.mil
jon.m.strother@nga.mil
kimberley.h.berger@nga.mil

Macromedia Flash is a trademark of Macromedia, Inc. ArcIMS, ArcMap, ESRI, and MOLE are trademarks or registered trademarks of Environmental Services Research Institute, Inc. JavaScript is a registered trademark of Sun Microsystems, Inc. Internet Explorer is a registered trademark of Microsoft, Inc. Netscape and Mozilla are registered trademarks of Netscape, Inc. All other product names may be trademarks or registered trademarks of their respective owners.