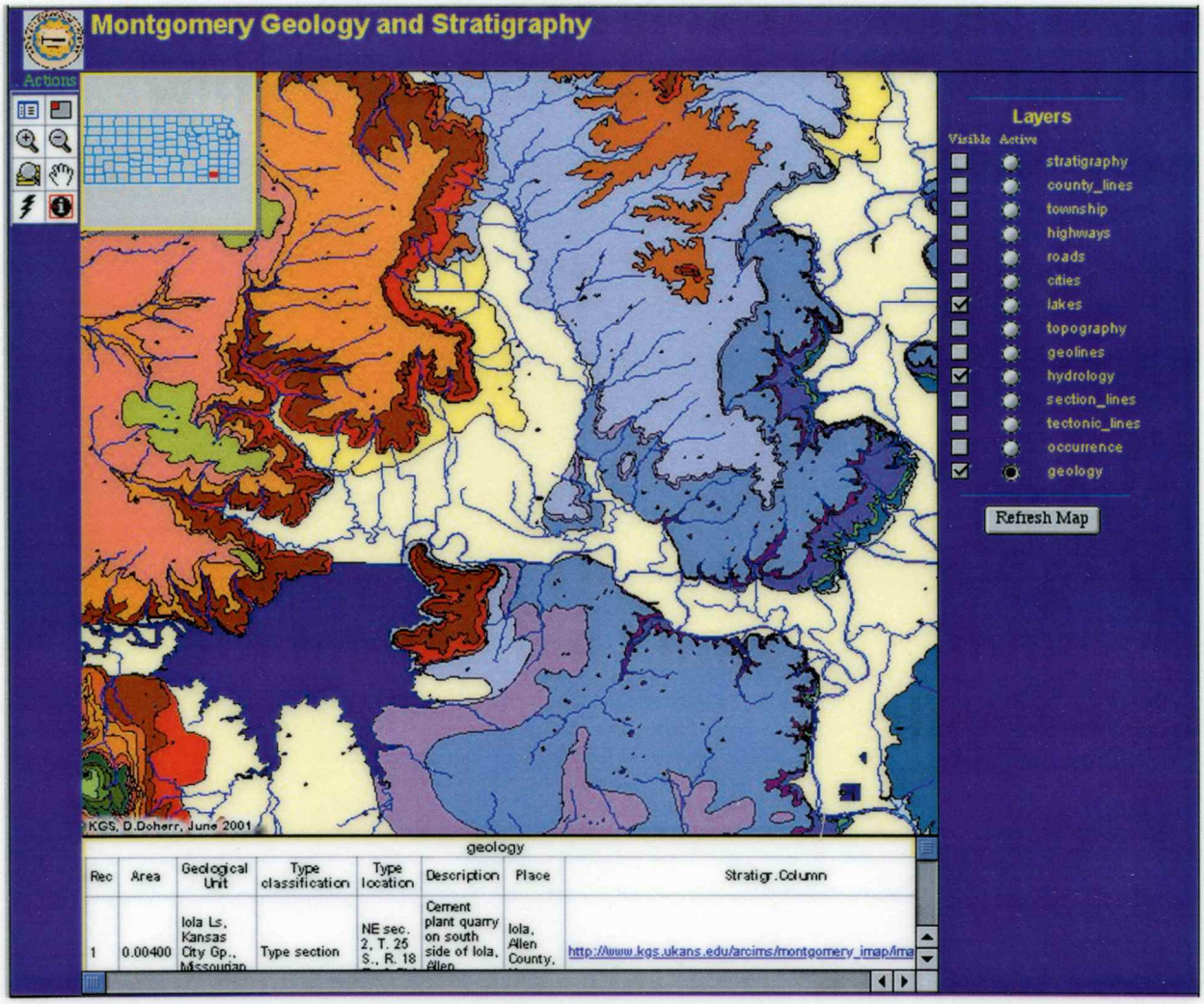
Implementing Online Geologic Maps with Interactive Links to Multimedia Archival Data

Detlev Doherr

Mathematical Geology Section
Kansas Geological Survey
1930 Constant Avenue
Lawrence, KS 66047



Open-File Report 2001-23

July, 2001

KGS Open-File Report 2001-23

Implementing Online Geologic Maps with Interactive Links to Multimedia Archival Data

A pilot project for Montgomery County, Kansas February – July, 2001

Detlev Doherr
Visiting Sabbatical Scholar
Mathematical Geology Section
Kansas Geological Survey
July 17, 2001

Prof. Dr. D. Doherr
Univ. of Applied Sciences
Badstrasse 24
D- 77652 Offenburg
GERMANY

Foreword

This report presents the results of my research at the Kansas Geological Survey in Lawrence, Kansas, from February to July 2001. I was in Lawrence as a visiting professor from Germany at the invitation of M. Lee Allison, Director of the Kansas Geological Survey (KGS), and John C. Davis, Chief of the Survey's Mathematical Geology Section. The opportunity for this visit came during a semester of sabbatical leave from my position as Chairman of the Computer Department at the University of Applied Sciences, Offenburg, Germany.

I appreciate very much the opportunity to work at the KGS and the experiences I have had during this time. I am satisfied that the results of my work at the Survey represent a positive contribution to the ongoing efforts of KGS staff. In this report I describe and document these results. Hopefully there will be further progress with interactive maps on the Internet after this first step. This will involve full integration of all available digital geologic maps with a broad range of archive databases, including the stratigraphic database demonstrated in this paper.

I gratefully thank my colleagues at the KGS, particularly John Davis for hosting my stay, David Collins and Gina Ross for their interest in my project and important advice and thematic discussions, Kurt Look for his support on Oracle, and Dana Adkins-Heljeson for assistance in handling the Apache webserver.

-- Detlev Doherr

Content:

1. Project idea

1. 1. Concept

Using ArcIMS (Arc Internet Map Service), Version 3.0 from ESRI it is possible, with care, to define interactive maps on the Internet. These maps use shape files from ArcInfo that include alphanumerical attribute data for each object on the map. After several discussions with geologists from the Kansas Geological Survey, a pilot project was formulated to develop an interactive map on the Internet using the Survey's UNIX based ArcIMS, with hyperlinks to related digital images and Internet access to an existing digital lexicon database of stratigraphic nomenclature. The digital geologic map of Montgomery County was chosen for the pilot project. This report addresses questions of how to access an Oracle based information system and how to retrieve data about a mapped geologic unit, selected by a user by pointing to a map object on an Internet browser.

With some experience in program development, the system can be modified to define a hotlink for a map layer, which activates a field from the attribute table as a hyperlink. The field should contain an URL with a website address or the name of an image on a web directory.

1. 2. Milestones

The following steps were identified as milestones for development of this pilot project:

- Verify the functionality of ArcIMS on UNIX
- Create an interactive map of Montgomery County
- Define parameter fields in the attribute tables
- Define parameters for data selection in Oracle database
- Verify PL/SQL as programming language for Oracle
- Develop the procedure package on Oracle
- Design the website for Oracle data output
- Define hotlinks in the interactive map
- Customize the interactive map system
- Test system
- Document system

1. 3. Development tools

The UNIX platform (Solaris 2.6) for this project runs on an Ultra Sparc II with 2 processors and 768 MB RAM. The Kansas Geological Survey uses an in-house system, GIMMAP (Geodata Information Management, Mapping and Analysis Package) for primary digital geologic map data capture and management, and for design and publication of geologic maps. ArcInfo is used for creating and managing the interactive geological map data, and the Oracle database management system is used for data storage. The complete platform for development, as shown in the following table, is provided by these software products:

ArcIMS 3.0
ArcInfo
Oracle 8.1.7
Apache Webserver (1.3.14)
Servlet Engine JRE 1.2.1

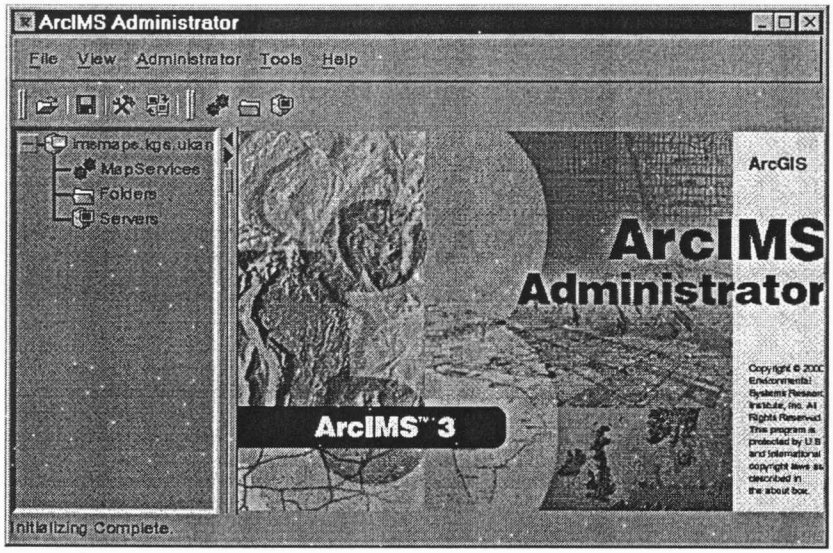
For system development the software was used to create several new files with source code and project data. These are listed in the following table together with the working steps.

Systems	Files/Data	Work
ArcView	Shape file	Create new attribute fields
ArcCatalog	Shape file, SDE	Create Geodatabase
AIMS-Author	AXL	Define shape files as layers for web service
AIMS-Admin	Map Service	Define and manage the Internet map services on the Virtual Server
AIMS-Designer	HTML, JPG	Webize the interactive map
Programming	PL/SQL, CGI, JAVA	Create Online Database on stratigraphic lexicon, define procedures on Oracle for online access by hyperlink, define queries

System platform at Kansas Geological Survey for development of interactive maps and database links

2. Using ArcIMS to produce interactive maps

The geological maps are available as shape files from ArcInfo. To produce interactive maps on the Internet, the package ArcIMS (*Arc Internet Map Server* by ESRI) is used. This is installed on the UNIX platform as version 3.0. The developer must be authorized to set the properties of the server output and the new map service and have permissions to use the shape files containing the data. In order to establish a new web service, the developer must have permissions to create a new directory on the webserver.



Start up screen from ArcIMS ,version 3.0 on UNIX

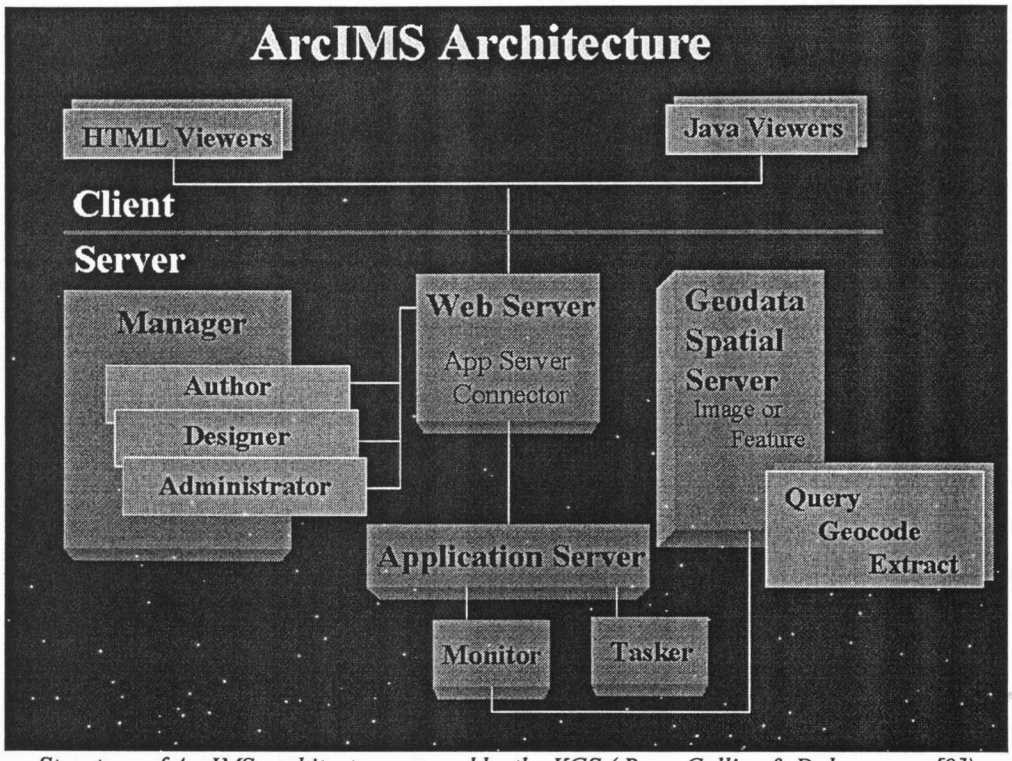
In other circumstances, or if the name conventions of shape files are not followed, the system may loop, and it may be impossible to establish a new map service. To produce an interactive map on the Internet, several steps are necessary:

1) The shape file from ArcInfo/ArcView must be described by an author file of type 'AXL'. The author file is created using the ArcIMS Author package.

2) A map service must then be defined for the virtual server, which is implemented by the system installation. This virtual server gets the request from the webserver and gives the map data back.

3) After the map service with the AXL file is created, it must be started by the ArcIMS Administrator package.

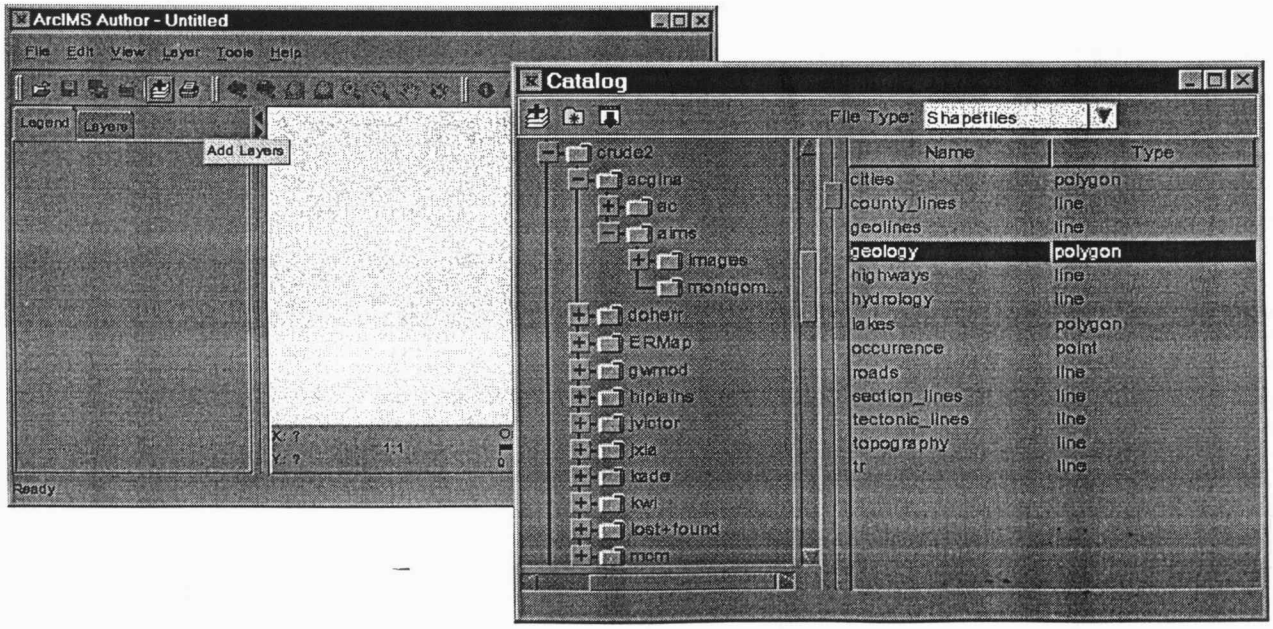
4) The ArcIMS Designer package uses the running map service to produce a standard website for the output as an interactive map.



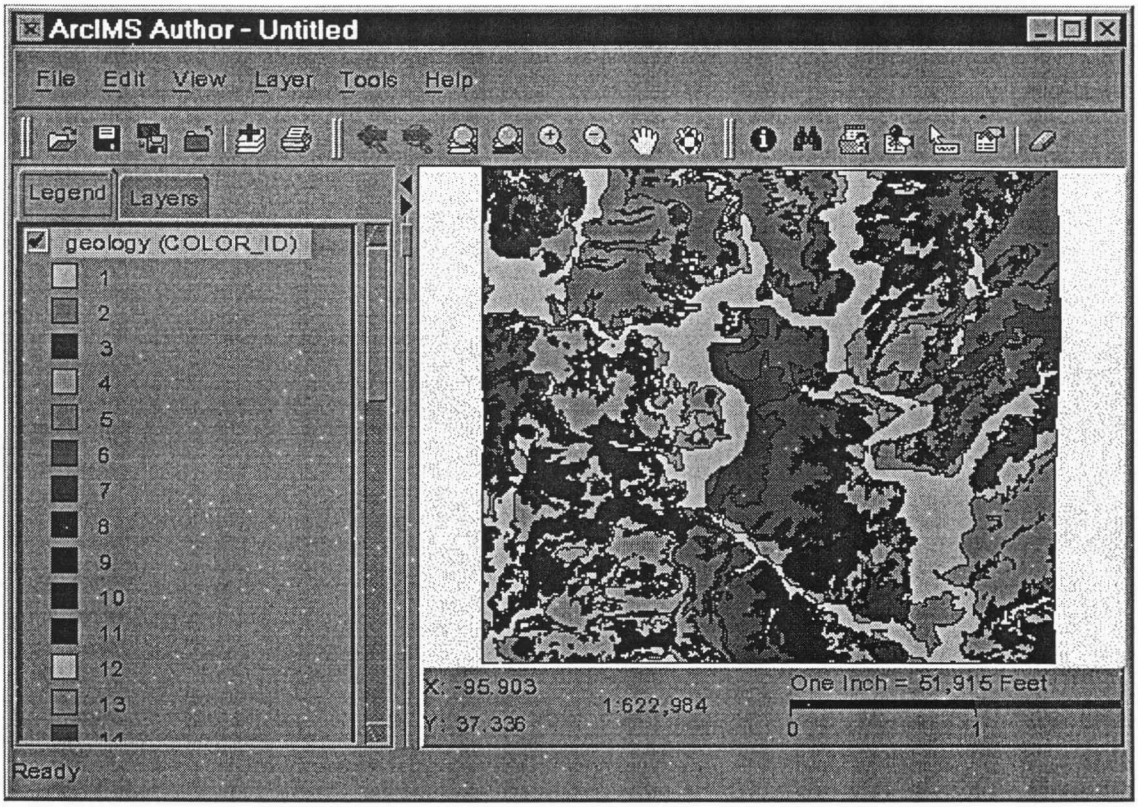
Structure of ArcIMS architecture as used by the KGS (Ross, Collins & Doherr , see [9])

2. 1. Creating an AXL file

The first step to define an interactive map is to create an AXL file from the available shape files using the AIMS Author package. This step is comparable to the definition of projects in ArcView. This file integrates, for the web service, all the path and name information about the layers and their objects. Only those layers listed inside the AXL file may be shown later in the web output. When the layers are defined in the AIMS Author package, the color or label definitions or sequence of layers can be customized by ArcIMS.



Creating the AXL file from available ArcView shape files, which can be shown on the interactive map



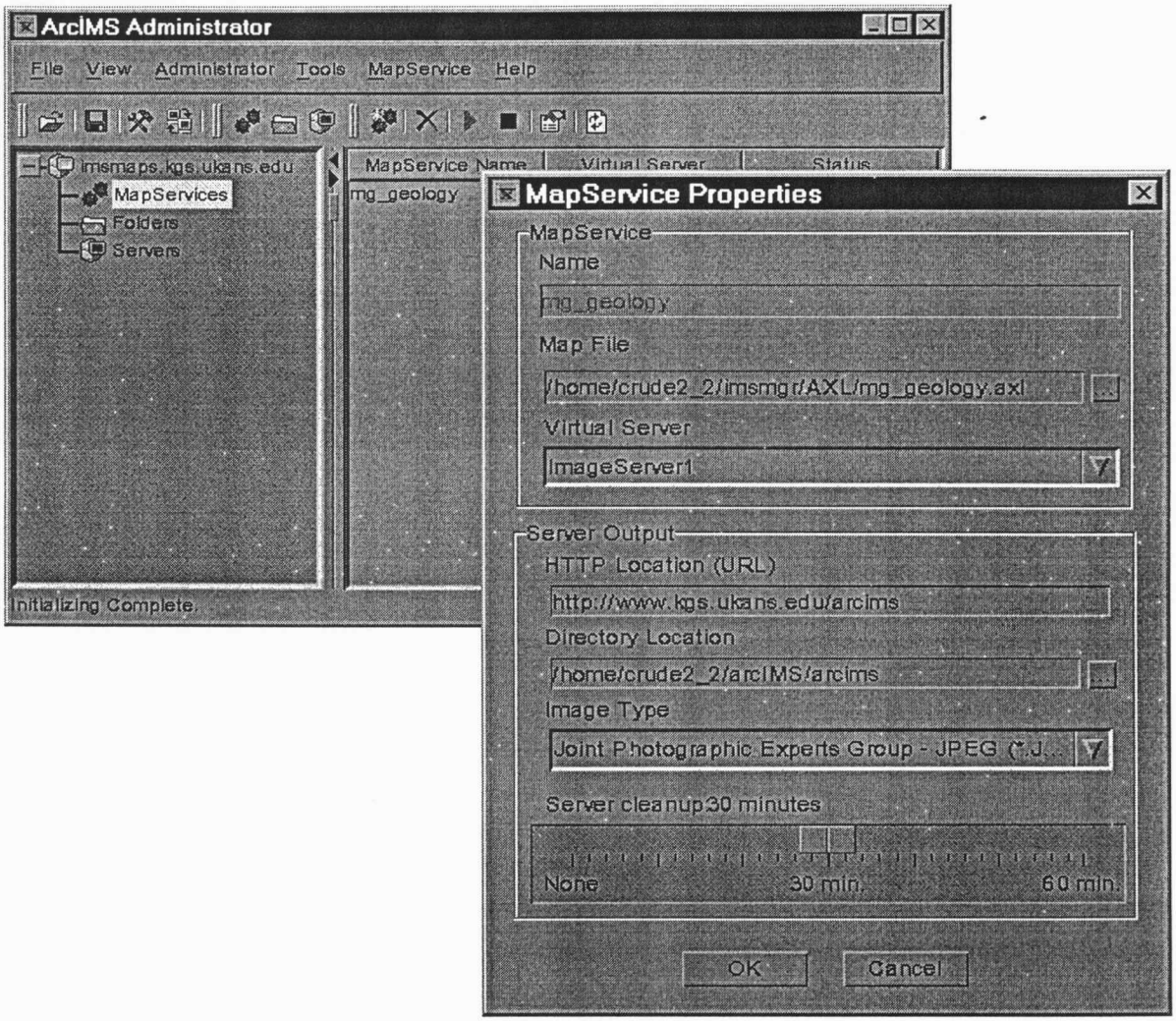
Using the AIMS Author package for preparing interactive map output of the virtual server, with definition of layers, labels, and colors of the map object.

The ArcIMS Author shows the shape files as layers and allows the presentation to be customized.

2. 2. Creating a map service

A map service is established on the virtual server from ArcIMS. The virtual server processes requests for maps and creates cartographic image files for Internet output. This virtual server may be connected with the webserver such as the Apache webserver, used by the KGS, and may use a variety of connectors, including Servlet, ColdFusion, or ActiveX connectors.

The Servlet Connector is the standard connector for ArcIMS and uses the ESRI's standard ArcXML language for communication with the webserver (ESRI : 'Using ArcIMS' , see [8])



Defining a map service from an existing AXL file for a virtual server 'ImageServer1' for webserver output at location 'http://www.kgs.ukans.edu/arcims'

Essentially, a map service provides information to the virtual server on how to manage a request for an interactive map. The properties of any single map service specify the AXL file to be used, the directory for internet access as a physical directory which is, at the same time, defined as an internet directory by the webserver as part of the URL. The JPEG format is the standard image type for virtual server output.
The cleanup time for the virtual server sets the running time of the map service on the virtual server. If this time is defined as null, the service would run forever unless stopped by the AIMS administrator.

The final property used for the Montgomery County project is:

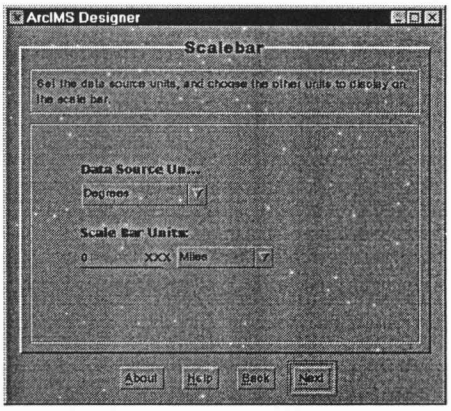
1) server output = 'http://www.kgs.ukans.edu/output',
2) directory location = ' /home/crude2_2/arcIMS/maps'.

2. 3. Designing the web page

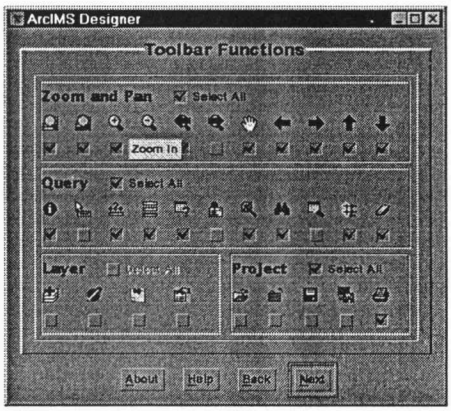
Once the map service is running, the AIMS Designer may be used to define the web page to be presented to the user via the Internet. This design may be customized later, but it is very helpful to create an initial standard design with ArcIMS Designer. It is not possible to change some of the definitions of the design by editing the web service directly. If there is any reason to change the design, there are two possibilities:

1) create a new web service and overwrite the old one,
2) edit and customize the Java script and Java parameter files, which are produced by ArcIMS.

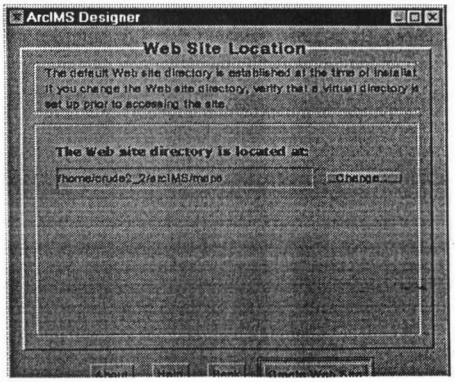
The standard design for the Montgomery County project contains:
A scale bar definition



A toolbar definition

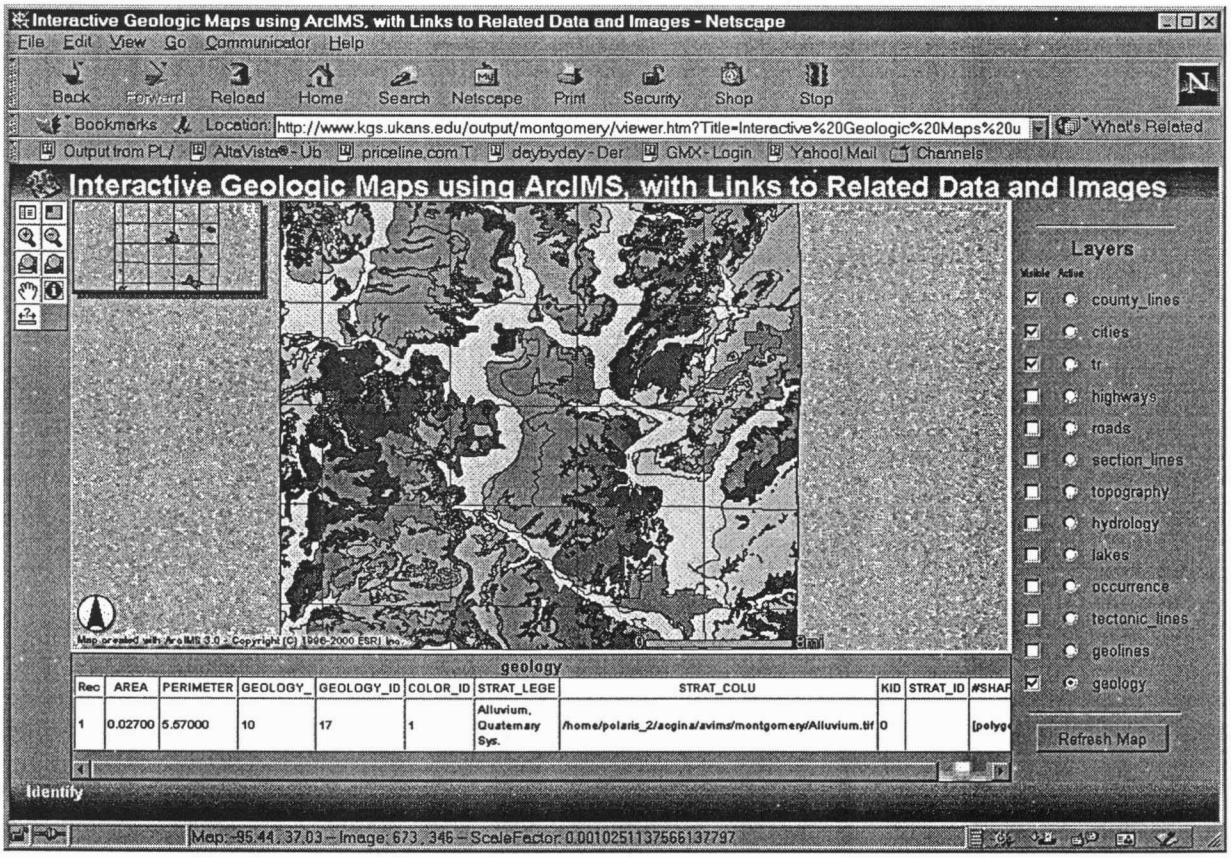


The web page location



and several other parameters which are described in 'Using ArcIMS' from ESRI [8]. The design of the frames of the web page, such as the top frame, map frame, toolbar frame and legend frame, are produced by ArcIMS following its standard definitions. To change these, the developer must edit the Java script and Java parameter files.

The standard design for the project is shown in the following image.



Internet output of the project map in the standard design for geological information on Montgomery county, with an overview map of the county and attribute table for the 'geology' layer

The web service for Montgomery County contains thirteen layers that can be activated or deactivated, and presented or not presented on the website. Only the active (and visible) layer is linked to the information button (i) in the left column of the action icons table. When this button is clicked, the user can click on a graphical object inside the map to get alphanumeric data from the attribute table for this specific unit, given in a table below the map. The data presented depends on the active layer and the marked object.
Using other action icons, the user may zoom in and out of the map, pan the map, return to a display of the whole project area, switch the layer list on the right column to the map legend, switch on or off the overview map on the upper left position of the map, or measure distances on the visible map.

The function of each icon in the left upper frame of the web page is shown in a small yellow box when the mouse pointer is moved on the icon. The functions are:

Show legend or list of layers
In the right frame a list of the available layers, or a legend for the maps is shown. A click on this icon switches from one to the other.

Overview map
An overview map is presented in the upper left corner of the map, no matter which map layer is active and visible. A click on this icon hides the overview map or shows it again.

Zoom in
Starting from the full extent of the Montgomery county map, the user can use the zooming function by clicking on the icon and dragging to define a zoom window on the visible map.

Zoom out
To zoom out, the user clicks on this icon and clicks on one part of the visible map.

Zoom to full extent
No matter which area of the visible map is shown, the full extent of the map is presented with a click of this icon.

Pan
Shifts the map area inside the visible window. This is done by clicking on the map area, then holding down the left mouse button and moving the cursor in the desired direction. The coordinates of the map objects are not changed by this function; only the presentation area is moved.

Hyperlink
On the active map, clicking on the hyperlink icon activates a hotlink between the map area and part of the attribute table that describes the object. Every single map object could have a link to another URL or to an image, which will automatically be presented by clicking on the map object..
The Montgomery County map contains hyperlink definitions which call the stratigraphic lexicon and present stratigraphic and citation data for each geological unit in the layer 'Geology'.

Identify

When this icon is highlighted, the user can click on any map object in the active map to get the attribute data for the selected object and further descriptions as alphanumeric data.
It is strongly recommended that the active layer be visible.

On the right side of the web page, either the legend or the layer list is shown. In the layer list, the user can choose the active and the visible layer. Only one layer can be active. To see the result of changes in the visibility of layers, it is necessary to 'Refresh the Map'.

Customizing the interactive map is described in chapter 4 of this report: 'Customizing the web page'.

3. Procedures on Oracle for lexicon data access

3. 1. Stratigraphic Lexicon Data

The Kansas Geologic Names Database is a digital stratigraphic lexicon. It contains information on the description and nomenclature history of named rock units in Kansas. It is structured as a relational database implemented in ORACLE. The database was derived from text files for Bulletin 231 of the Kansas Geological Survey, *Lexicon of geologic names of Kansas (through 1995)*.. It incorporates a multitude of revisions, corrections and additions to the printed publication. The database currently resides as a schema (LEXICON) in an Oracle 8.1.7 database. The database is designed and edited by D. R. Collins, with Oracle implementation by K. K. Look, both of the Kansas Geological Survey.

One of the main purposes for developing Internet access to the Kansas Geologic Names Database is to present the information contained in this database related to rock units featured on geologic maps or cross sections being served as interactive maps on the Internet. Stratigraphic intervals shown on a geologic map may represent one or more named rock units. From one map to another the grouping of units within mapped intervals may change. Providing a useful link of digital geologic maps to the KGN database requires a database structure that permits correlation of mapped intervals on any map with the corresponding formally accepted nomenclature for included rock units.

Tables defined in the Kansas Geologic Names Database on ORACLE:

AUTHORS
CITATIONS
LOOKUPS
OCCURENCES
OCCURENCES_IMAGES
OCCURENCES_SECTIONS
ORGANIZATIONS
PROJECTIONS
SEQUENCES
SEQUENCES_DETAILS
SOURCES
SOURCES_AUTHORS
SOURCES_RELATIONSHIPS
UNITS
UNITS_RELATIONSHIPS

For the current Internet presentation of these stratigraphic data, only some of the existing data tables are necessary. Geologic maps, and numerous published reports, present information on

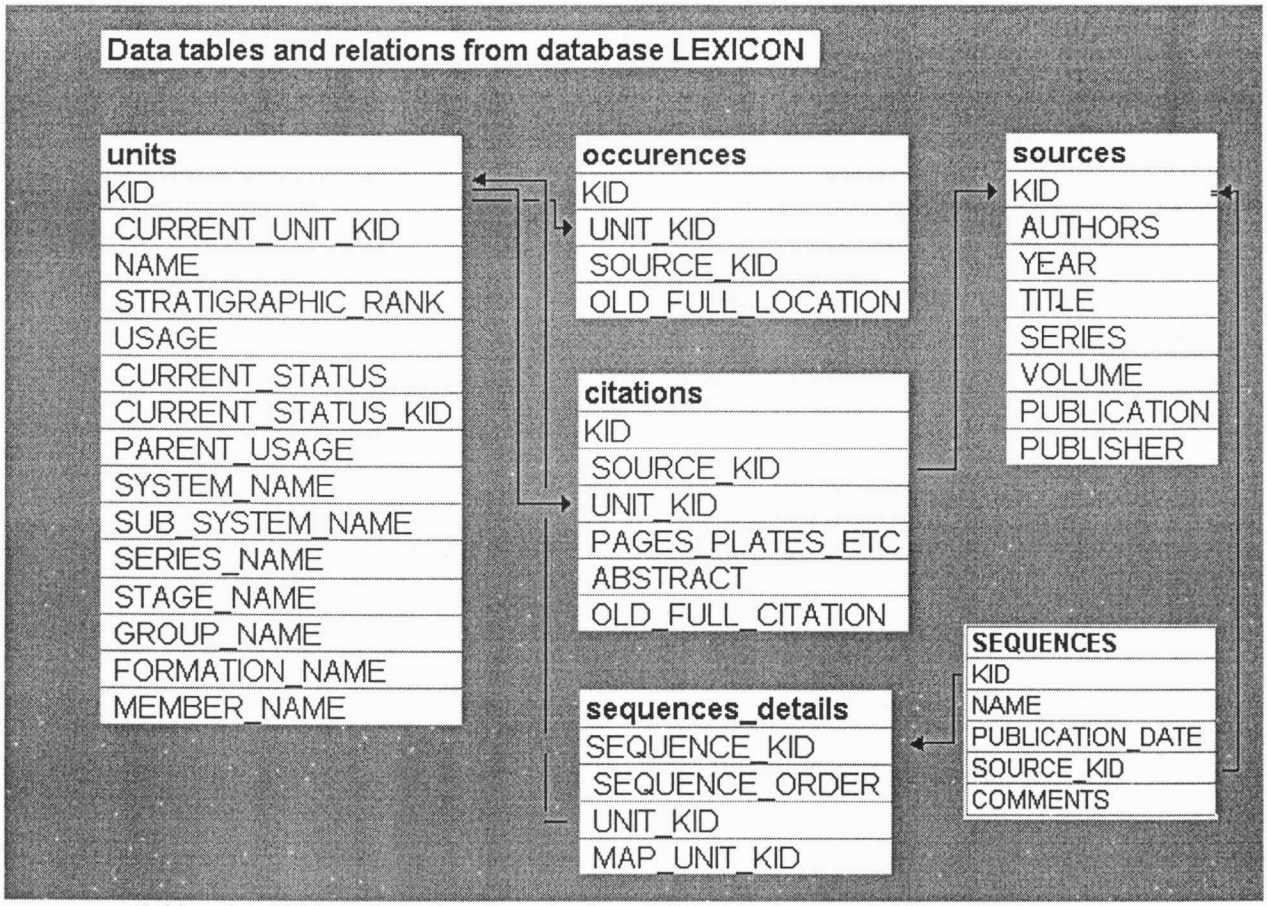
stratigraphic sequences representing the geologic interpretations at the time of publication. These sequences are identified in the SEQUENCES table with a unique KID, NAME, and SOURCE_KID ('KID' is the acronym for the unique 'Kansas IDentification numbers' used in ORACLE databases of the Kansas Geological Survey).
The SEQUENCES_DETAILS table is a correlation table defining many-to-many relationships between sequences, mapped units, and recognized geologic unit names. Within the SEQUENCES_DETAILS table, in the case of geologic maps, the SEQUENCE_KID for the map is associated with unique MAP_UNIT_KIDs representing each of the geologic units or intervals represented on the map. Where the mapped interval is a recognized stratigraphic unit (member, formation, group, etc.) there will be one record for that mapped interval on that map with the SEQUENCE_KID, MAP_UNIT_KID, and UNIT_KID. The mapped interval is a grouping of geologic units recognized as present in the area by the author, but not shown in the detail of the map. In this case there will be multiple records with the same SEQUENCE_KID and MAP_UNIT_KID but a unique UNIT_KID for each of the units within the mapped interval.

The structure of the data tables is important for the several select commands, which are included in the program. To find the citations for a specific mapped unit, the select command must find the one or more records in the CITATIONS table containing the UNIT_KID identifiers from the SEQUENCES_DETAILS table according to the MAP_UNIT_KID and the SEQUENCE_KID parameters. . There could be multiple UNIT_KIDs found with a single query. For each UNIT_KID there will be a different data set selected from the CITATIONS table.

Sometimes there are no data available for the given UNIT_KID because of differences in the nomenclature used by the author and the currently accepted nomenclature for a geologic unit. Then the content from the data field CURRENT_UNIT_KID from the table UNITS should be used to find the information about citations and authors.

The key that relates the UNITS table to the CITATIONS table is the match between the KID of the UNITS table and the UNIT_KID of the CITATIONS table. Similarly, the SOURCES table is related to the CITATIONS table by the KID of the SOURCES table and the SOURCE_KID of the CITATIONS table.

Type localities of rock units are available from the table OCCURENCES, which is related to the table UNITS by UNIT_KID. The current implementation does not make full use of the information contained in the OCCURENCES table, but used instead the simplest query. This will be modified along with other extensions in the future.



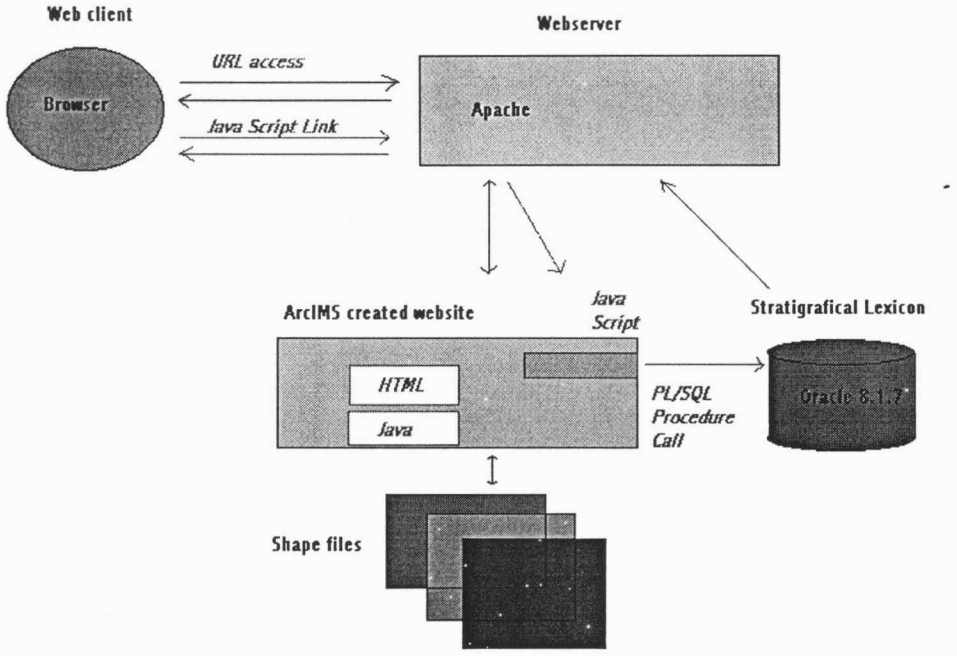
Tables, data fields and relations from database LEXICON, that are significant for this project. The arrows show the relationship between different tables.

The structure of the data tables is important for all the select commands, which are included in the program for online database presentation and for defining hyperlink access.

For example to get information about UNIT_KID for a geological unit in a specific map, a query is needed to the table SEQUENCES_DETAILS, which selects UNIT_KID for a given SEQUENCE_KID and MAP_UNIT_KID. There could be multiple UNIT_KIDs found with a single query. For each of these a different data set could be selected from the CITATIONS table.

3. 2. Establishing the link between an interactive map and the lexicon data

The shape files of geological units contain a variety of alphanumerical attribute data about the polygonal limited areas of the map. This structure can be used to add further fields to the table, including one or more identifiers for the Oracle database. Using ARCIMS as described here, a web presentation and web page design can be developed as an interactive geological map with several layers, with attribute data for all geological areas and a query function to present these attribute data in the browser. To link to the Oracle database, a module in PL/SQL has been designed that takes the identifier(s) from the attribute field, connects to the Oracle system, and selects the stratigraphic data. A new browser window is opened to display the stratigraphic descriptions.



Concept for a link between an interactive map and the Oracle database with PL/SQL procedures and a Java hotlink to the URL address of the procedure package of Oracle

3. 3. Using the PL/SQL Developer Platform

The ORACLE database includes a procedure package named PL/SQL (See: ORACLE Application Server- PL/SQL Web Toolkit [4]). This package contains a programming platform based on a procedural language, that offers several different possibilities for data access using SELECT commands, data manipulation, and data presentation in web pages. These commands are very close to the HTML programming language (**Musciano, C. & Kennedy, B. (1997)**: HTML- The Definitive Guide.- [3]). To generate HTML tags, there are two packages available:

1) the htp (hypertext procedures),
2) the hft (hypertext functions) package.

The difference between them is that the hft function passes its output to its caller.

For example:
To create a web page some HTML tags for the HEAD and the BODY are needed. In PL/SQL these are given by using the hpt package.

HTML	HTP
<HEAD>	Htp.HeadOpen;
</HEAD>	Htp.HeadClose;
<BODY>	Htp.BodyOpen (background,attributes);
...	...
</BODY>	Htp.BodyClose;

Htp functions from PL/SQL defining the structure of a web page

The ORACLE cartridge PL/SQL is the development platform for the procedural language and corresponds to the database access descriptor (DAD) (See: ORACLE Application Server-Developer's Guide [6]). This enables the Web user to invoke stored procedures via browser. These procedures can retrieve data from the database, select information and return the included data to web pages.

The PL/SQL allows the developer to use the POST and GET methods, which instruct the browser how to pass parameter data to the procedures. The POST method passes the parameters in the requested body, whereas the GET method passes the parameters in a query string.

Packages of procedures contain one or several procedures and functions. There is a fixed structure for a package, beginning with the declaration part, that lists all procedures and functions inside the package, followed by the body with the programming code.

```
CREATE PACKAGE test
        AS
        PROCEDURE test_1 (val IN varchar2);
        PROCEDURE test_2 (val IN varchar2);
END test;
CREATE OR REPLACE PACKAGE BODY test AS
/*      Package body with 2 procedures      */
PROCEDURE test_1
        (val IN varchar2)
        is
        BEGIN
        ....
END test_1;
PROCEDURE test_2
        (val IN varchar2)
        is
        BEGIN
        ....
END test_2;

/*      End package body                    */
END test;
```

The example shown gives the structural elements for a package named 'TEST', which has two integrated procedures. Both procedures use a parameter named 'val', which must be added to the procedure call. BEGIN and END define the frame for the program code inside each procedure.

The programming system is integrated in the SQLPLUS environment from ORACLE. To develop a PL/SQL package, you should start ORACLE with the following environment variable settings:

ORACLE_HOME:	setenv ORACLE_HOME /home/polaris_2/app/oracle/product/8.1.7
ORACLE_SID :	setenv ORACLE_SID ABYSS
PATH :	setenv PATH $PATH':'$ORACLE_HOME/bin

After starting SQLPLUS, you can store the procedure package in a home directory of the connected UNIX system, edit and implement the package using a common editor like VI, and compile the package in SQLPLUS by calling the PL/SQL compiler with '@' and the entire package name (such as 'lexicon.sql'). If there are no errors in coding, you can access the package directly by an Internet browser, using the URL from the ORACLE server and the package name. For the LEXICON program, the address is:

http://polaris.kgs.ukans.edu/pls/abyss/autocarto.lexicon.lexicon

The whole package contains not only the procedure for the link to the interactive map, but also represents a complete online presentation of the stratigraphic lexicon with several functions and options for data selection. For details, see chapter 3.5 of this report.

In the appendix the source code of the map link procedure is listed.

3. 4. Program Structures and Procedures

The new program package is called 'LEXICON', which contains the following procedures and functions on PL/SQL. This is developed for the user 'Autocarto' at the computer 'abyss'.
In the following chapters every procedure and function is described and the select commands are listed.

3. 4. 1. Procedures

WEBHEADER
Defines the header of the output web page.

WEBFOOTER
Defines the bottom of the output web page.

LEXICON
Defines the first web page and presents some radio buttons for selecting a specific function.
Calls WEBHEADER and WEBFOOTER, sends the content of variable BU to the procedure FUNCSEL.

FUNCSEL
Takes the value of variable BU and calls one of the procedures, depending on the value of BU.
Calls SEL_TABS, SEL_KID_NAME, SEL_MAPUNITKID, SEL_AUTHOR, SEL_HELP

PR_MAINDATA

Reads parameter 1 as 'TEXT', parameter 2 as 'LEX_TEXT'
Prints out the given values in TEXT and LEX_TEXT on web side, if there is a value to be found.
Called by procedure SHOW_DATAHEAD and SHOW_DATABODY.

SEL_UNIT_KID

Reads parameter 1 as 'P1' and parameter 2 as 'P2'.
Reads data from database for cursor UNIT, cursor LEX
Defines counter for restriction of amount of data output lines to a maximum of 1000 lines.
Calls WEBHEADER, WEBFOOTER, SHOW_DATAHEAD, SHOW_DATABODY

SHOW_LEXICON

Selects all data from given SEQUENCE_KID, parameter P1 is a selected KID for a unit.
Reads parameter 1 as 'P1'.
Reads data from database for cursor SK, SK_NO_CITATION.
Calls WEBHEADER, Function SEL_CUKID_FROM_KID, SHOW_DATAHEAD, Function
SEL_OLD_FULL_LOCATION_FOR_UKID, SHOW_DATABODY, WEBFOOTER.
Defines PARA1 as 'select count(c.UNIT_KID) into LOCAL_KID_NUMBER from
LEXICON.CITATIONS c, LEXICON.UNITS u
where c.UNIT_KID=u.KID
and u.KID = PARA1;'.

SEL_TABS

Shows all data about the defined data tables in LEXICON.
Reads data from database for cursor CTEST
Calls WEBHEADER, WEBFOOTER.

SEL_HELP

Shows the use of the query system.
Calls WEBHEADER, WEBFOOTER.

SEL_KID_NAME

Shows web page and gives a list of all KIDS with the possibility of selection.
Calls WEBHEADER, CHOOSEN_KID, CHOOSEN_NAME, WEBFOOTER.
Reads data from database for cursor KIDS, NAM.

CHOOSEN_KID

Presents selected field inside the web side, produced by SEL_KID_NAME.
Reads data from parameter 1 as 'SKID'.
Gets information from procedure SEL_KID_NAME.

CHOOSEN_NAME

Selects the UNIT_KID for selected geological unit name.
Reads data from parameter 1 as 'SNAME'.
Reads data from database for cursor K4N.
Gets information from SEL_KID_NAME.
Calls SHOW_LEXICON or WEBHEADER, WEBFOOTER.

SEL_AUTHOR

Selects single Author and searches data.
Reads data from database for cursor AUT.
Calls WEBHEADER, CHOOSEN_AUTHOR, WEBFOOTER.

CHOOSEN_AUTHOR

Calls for authors and data inside the web page.
Reads data from parameter 1 as 'NAME'.
Reads data from SEL_AUTHOR.
Calls SHOW_AUTHOR.

SHOW_AUTHOR

Presents authors and data on web page.
Reads data from parameter1 as ' ANAME'.
Reads data from database for cursor SK.
Calls WEBHEADER, WEBFOOTER, PR_MAINDATA, SHOW_DATABODY.

SEL_MAPUNITKID

Selects SKID and Map_UNIT_KID from data tables.
Reads data from data base for cursor SKID.
Calls WEBHEADER, WEBFOOTER, CHOOSEN.DETAILS.

CHOOSEN_DETAIL

Gets the SKID information from list.
Reads data from parameter 1 as 'SELSKID'.
Reads data from data base for cursor SSKID.
Calls WEBHEADER, SEL_MAPUNITKID, WEBFOOTER.

CHOOSEN_MUKID

Definition of a list for selecting Map_UNIT_KID.
Reads data from parameter 1 as 'MUKID'.
Gets information from CHOOSEN_DETAIL.
Calls SEL_SKID_FROM_MAPUNITKID.

SHOW_DATAHEAD

Shows the selected lexicon information for data head.
Reads data from parameters 1 to 8 as 'S_NAME', 'S_USAGE', 'S_PARENT_USAGE', 'S_STAGE_NAME', 'S_SERIES_NAME', 'S_SYSTEM_NAME', 'S_NAME_ORIGIN', 'S_GEO_EXTENT_SURFACE'.
Calls PR_MAINDATA.

SHOW_DATABODY

Shows the selected lexicon information for data body.
Reads data from parameters 1 to 8 as 'S_AUTHORS', 'S_YEAR', 'S_TITLE', 'S_PUBLICATION', 'S_PAGES_PLATES_ETC', 'S_PUBLISHER', 'S_ABSTRACT', 'S_OLD_FULL_LOCATION'.

SEL_SEQUENCE

Select single Sequence from SEQUENCE_DETAILS and reports citations for each unit which has a UNIT_KID.
Reads data from database for cursor SQKID.
Calls WEBHEADER, CHOOSEN_SEQKID, WEBFOOTER;

CHOOSEN_SEQKID

Call for authors and data inside a webside.
Reads data from SEL_SEQKID.
Calls SHOW_SEQUENCE.

SHOW_SEQUENCE

Presents all units and citations on a single sequence.
Reads data from database for cursor SK.
Calls WEBHEADER; PR_MAINDATA, SHOW_DATABODY, WEBFOOTER.

3. 4. 2. Functions

Function SEL_NAME_FROM_UKID

Function for selecting a NAME from UNITS where KID is given.
Reads data from parameter 1 as 'UKID'
Returns SELECT_NAME as varchar2.

Function SEL_SKID_FROM_MAPUNITKID

Function for selecting the SEQUENCE_KID from SEQU_DETAILS for a given MAP_UNIT_KID.
Reads data from parameter 1 as 'MUKID'.
Returns SELECT_SKID as varchar2(40).

Function SEL_CUKID_from_KID

Function for searching a CURRENT_UNIT_KID from UNITS, which contains an alternative KID for a geol. unit; if so, select data for this instead of the KID.
Reads data from parameter 1 as 'SKID'.
Returns SELECT_CUKID as varchar2(40).

Function SEL_OLD_FULL_LOCATION_FOR_UKID

Function for preselecting the value of the OLD_FULL_LOCATION and returns this as a string variable or a zero value string.
Reads data from parameter 1 as 'PARA1'.
Returns SELECT_OLDLOC as varchar2(4000).

3. 4. 3. Selects

Procedure SEL_UNIT_KID

```
cursor UNIT is
  select UNIT_KID
       from LEXICON.SEQUENCES_DETAILS
       where SEQUENCE_KID = P1
       and MAP_UNIT_KID = P2;
Cursor LEX is
  Select U.NAME, U.USAGE, U.PARENT_USAGE, U.SYSTEM_NAME, U.SERIES_NAME, U.STAGE_NAME,
       U.NAME_ORIGIN, U.GEO_EXTENT_SURFACE, C.ABSTRACT, C.PAGES_PLATES_ETC, S.AUTHORS,
       S.YEAR, S.TITLE, S.PUBLICATION, S.PUBLISHER
       from LEXICON.UNITS U, LEXICON.CITATIONS C, LEXICON.SOURCES S
       where U.KID = UNIT_REC.UNIT_KID
       and U.KID=C.UNIT_KID
       and C.SOURCE_KID = S.KID;
```

Procedure SHOW_LEXICON

```
cursor SK is
  Select U.NAME,U.USAGE,U.PARENT_USAGE,U.SYSTEM_NAME,U.SERIES_NAME,U.STAGE_NAME,
       U.NAME_ORIGIN,U.GEO_EXTENT_SURFACE,C.ABSTRACT,C.PAGES_PLATES_ETC,S.AUTHORS,
       S.YEAR,S.TITLE,S.PUBLICATION,S.PUBLISHER
       from LEXICON.UNITS U,LEXICON.CITATIONS C, LEXICON.SOURCES S
       where U.KID = PARA1
       AAND C.SOURCE_KID = S.KID
       AND U.KID=C.UNIT_KID;
cursor SK_NO_CITATION is
  select NAME,USAGE,PARENT_USAGE,SYSTEM_NAME,SERIES_NAME,STAGE_NAME,
       NAME_ORIGIN,GEO_EXTENT_SURFACE
       from LEXICON.UNITS
       where KID = PARA1;
```

Procedure SEL_TABS

```
cursor CTEST is
  select OWNER, TABLE_NAME
       from ALL_TABLES
       where OWNER='LEXICON'
       order by OWNER;
```

Procedure SEL_KID_NAME

```
cursor KIDS is
  select distinct KID from LEXICON.UNITS
       order by KID;
cursor NAM is
  select distinct NAME from LEXICON.UNITS
       order by NAME;
```

Procedure CHOOSEN_NAME

```
cursor K4N is
  select KID from LEXICON.UNITS
       where NAME = SNAME
      order by KID;
```

Procedure SEL_AUTHOR

```
cursor AUT is
  select distinct (substr (AUTHORS,1,80)) AUTH_50  from LEXICON.SOURCES
       order by AUTH_50;
```

Procedure SHOW_AUTHOR

```
cursor SK is
  select U.USAGE,U.PARENT_USAGE, C.ABSTRACT,C.PAGES_PLATES_ETC,S.AUTHORS,
       S.YEAR,S.TITLE,S.PUBLICATION,S.PUBLISHER
       from LEXICON.UNITS U,LEXICON.CITATIONS C, LEXICON.SOURCES S
       where S.AUTHORS = ANAME
       and C.SOURCE_KID = S.KID
       and U.KID=C.UNIT_KID;
```

Procedure SEL_MAPUNITKID

```
cursor SKID is
  select distinct SEQUENCE_KID from LEXICON.SEQUENCES_DETAILS
       order by SEQUENCE_KID;
```

Procedure CHOOSEN_DETAIL

```
cursor SSKID is
  select distinct MAP_UNIT_KID from LEXICON.SEQUENCES_DETAILS
        where SEQUENCE_KID=SELSKID;
```

Procedure SEL_SEQUENCE

```
cursor SQKID is          select distinct SEQUENCE_KID  from
LEXICON.SEQUENCES_DETAILS             order by SEQUENCE_KID;
```

Procedure SHOW_SEQUENCE

```
       cursor SK is          select S.SEQUENCE_KID, u.KID,
u.name, u.usage, u.parent_usage, u.system_name,                    u.stage_name,
u.name_origin, u.geo_extent_surface, c.abstract,
       c.pages_plates_etc, s.authors, s.year, s.title, s.publication, s.publisher         from
LEXICON.UNITS u, LEXICON.SEQUENCES_DETAILS s, LEXICON.CITATIONS c,
       LEXICON.SOURCES s             where s.UNIT_KID = u.kid             and u.KID=c.UNIT_KID
       and c.source_kid = s.kid              and s.SEQUENCE_KID = SEQKID          order by
s.SEQUENCE_ORDER;
```

Function SEL_NAME_FROM_UKID

```
       select NAME into SELECT_NAME
              from LEXICON.UNITS
              where KID=UKID;
```

Function SEL_SKID_FROM_MAPUNITKID

```
       select distinct SEQUENCE_KID into SELECT_SKID
              from LEXICON.SEQUENCES_DETAILS
              where MAP_UNIT_KID=MUKID;
```

Function SEL_CUKID_from_KID

```
       select CURRENT_UNIT_KID into SELECT_CUKID
              from LEXICON.UNITS
              where KID=SKID;
```

Function SEL_OLD_FULL_LOCATION_FOR_UKID

```
       select O.OLD_FULL_LOCATION into SELECT_OLDLOC
              from LEXICON.UNITS U, LEXICON.OCCURENCES O
              where U.KID = PARA1
              and U.KID=O.UNIT_KID;
```

3. 4. 4. Package structure

The different procedures and functions inside the package are organized according to the aspects of data selection in the Oracle database. This requires an extensive amount of program lines, but is helpful for further developments.

The web user chooses the kind of selection (LEXICON , FUNCSEL) for the data presentation by using the radio buttons on the first web page. Then he can choose items from the displayed listing fields for the name of a geological unit (SEL_KID_NAME), the identifier of specific map units (SEL_MAPUNITKID), or authors (SEL_AUTHOR). Listing fields present all the results of a query in procedures, which select data from a defined table. All items of the displayed listing field on a web page can be marked as parameters for further select statements in the following Oracle procedures.

The web output for the selected data is presented as a table, produced by several Oracle procedures (WEBHEADER, SHOW_LEXICON, DATAHEAD, DATABODY, WEBFOOTER). The procedure PR_MAINDATA formats the text output for this output table.

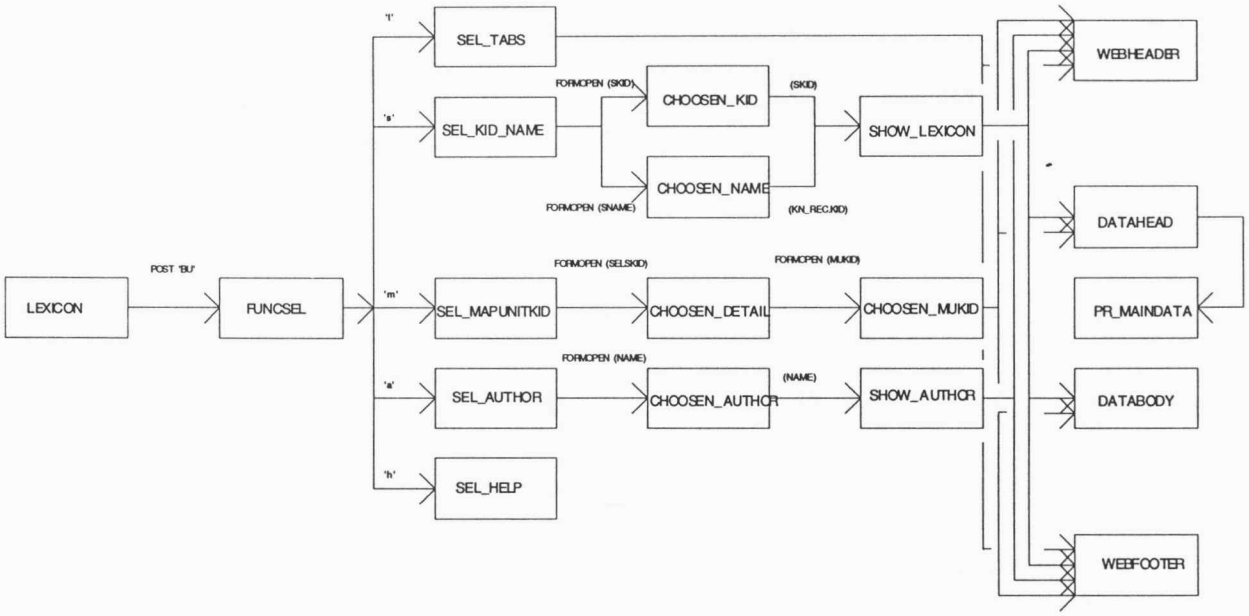
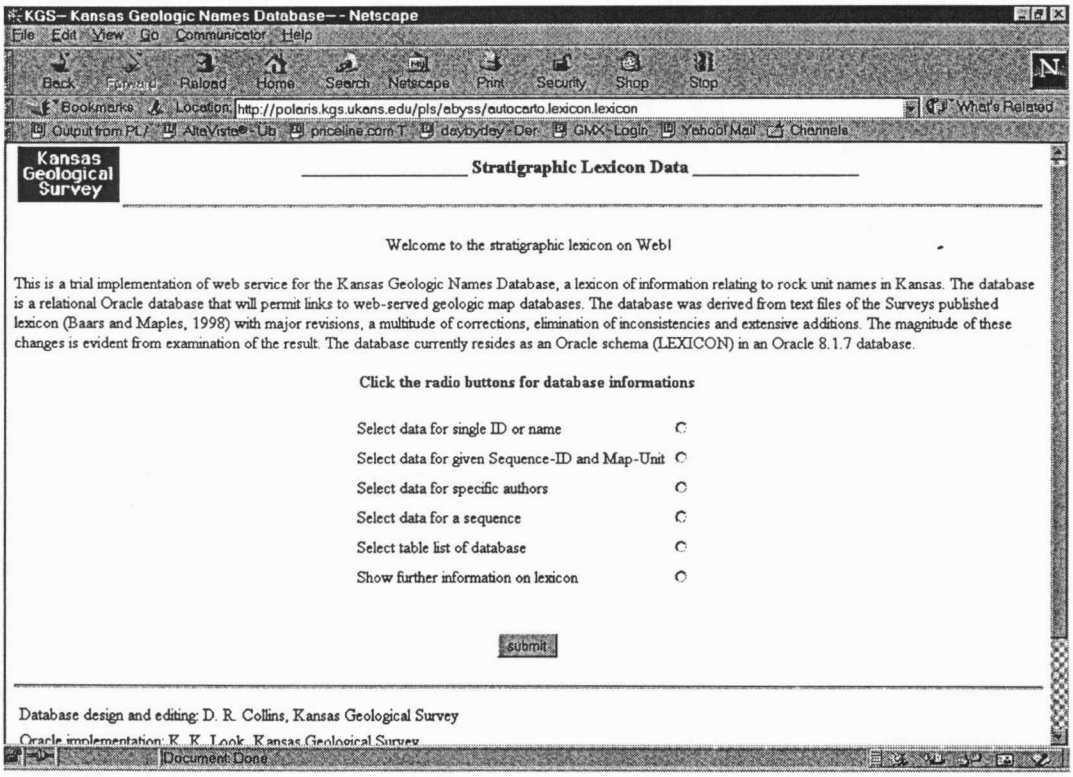


Diagram of the structure of the PL/SQL program and procedures which create output from the Oracle database as a web page. Arrows give the connection between procedures using parameters as noted above the arrows. For further explanation, see text.

3. 5. Online Data retrieval

Using the web address of the ORACLE procedure, it is possible to query the lexicon database directly by an Internet browser. The lexicon package contains a main web page with several radio buttons for function selection by the user.

The procedures for selecting and presenting the stratigraphical data are called easily by a click on one of these radio buttons and following a 'Submit'.



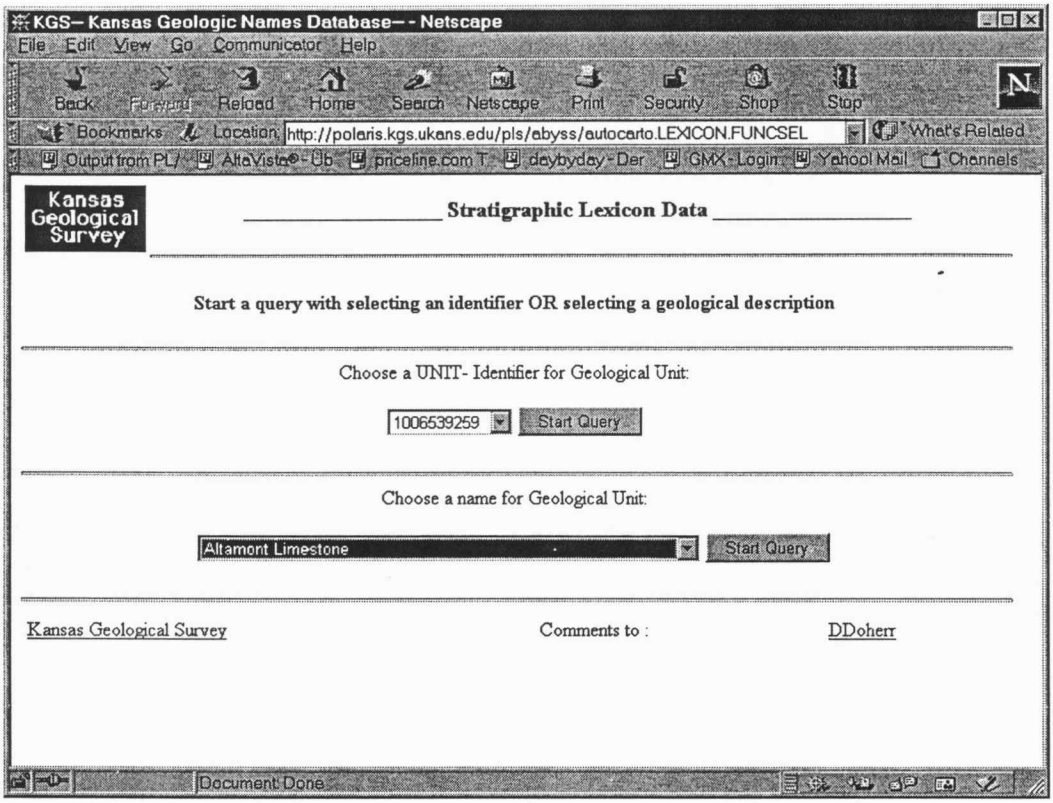
Web output of the starting page of the lexicon system, produced by Oracle procedures with several function calls

The following functions are implemented:

- **select data for single ID or name**

  For a defined Unit identifier or a selected unit name from a displayed listing field, all information about citations are shown.

  The selection of a parameter (ID or name) is very easy because the listing field contains preselected database information about all existing names or identifiers stored in the data base. The user can choose one of these items by mouse click. The query runs when the user clicks the button 'Start Query'.



Selection of data with UNIT identifier or the name of any geological unit using select field on the web page. The query is started by clicking on the button

- **Select data for given Sequence-ID and Map-Unit**
  After defining a sequence identifier and a map unit identifier, the data for the corresponding rock unit and the citations are shown.

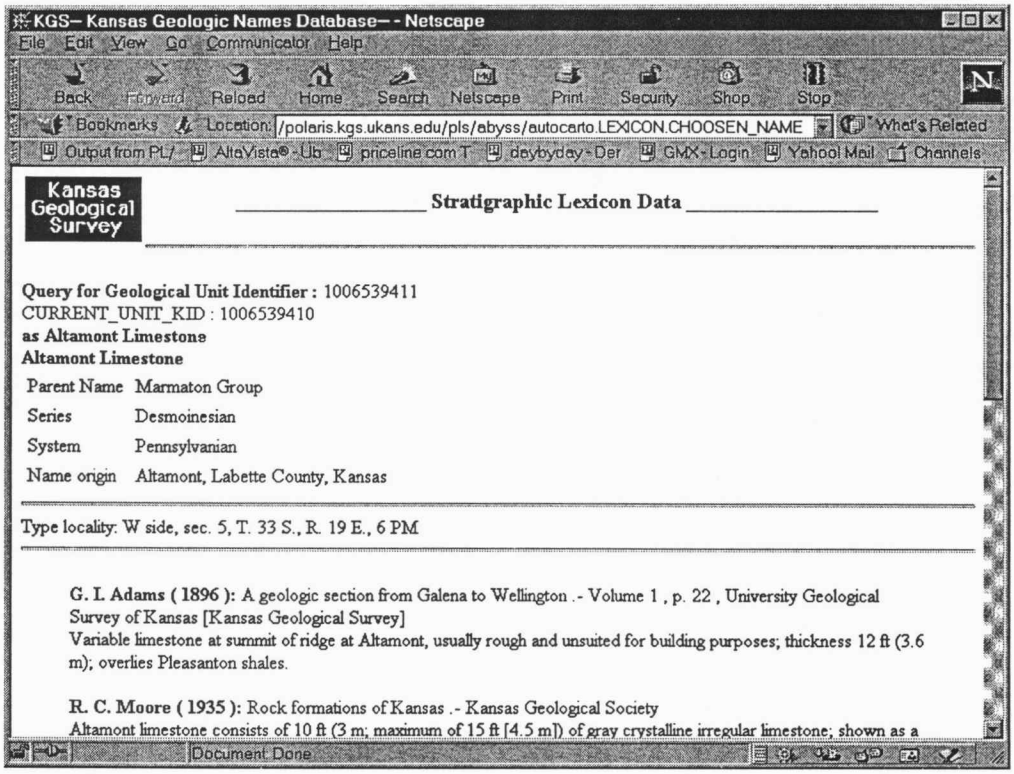
- **Select data for specific authors**
  All authors are presented as a listing from the database. The user can choose one of them to obtain all citations.

- **Select data for a sequence**
  The user can choose a single sequence identifier and get all data about integrated rock units and the corresponding citations as a list.

- **Select table list of database**
  A list of all the available data tables inside the lexicon database is displayed.

- **Show further information on lexicon**
  A help system is presented with further program details and description of the lexicon procedures from ORACLE.

After selecting of data in the database, web output is formatted as a data table and presented to the user. As an example, the image below shows a query result and the web output. The complete results can be viewed by scrolling in the opened browser window.



Presentation of a citation list after a query for a specific identifier for a geological unit

The performance of this web presentation is quite suitable, because there is little difference in loading time between common static web pages and dynamically produced web outputs of Oracle system. To accommodate the memory limitations of common browsers, the number of rows of data for citations for each selected unit is limited.

3. 6. Further Requirements

This package for an online database on the web, can be developed further by implementing more selection criteria and reports about the stratigraphic data. Because of the limited time for the research during the semester, only procedures for linking to the interactive maps and some selection functions were implemented. The source code for these procedures is given in the appendix.

Further requirements include:

1) Selection of a specific name (such as 'Douglas') where the stratigraphic rank such as 'Group', 'Member', or 'Formation' is given in the data field STRAT_RANK in table UNITS.

2) List of citations, when there is content in the field CONTAINED_IN_KID from the SOURCES table. That means there is no citation for the chosen unit but there is a citation which contains this unit. It is possible to implement a citation list with 'in:' that will yield all records from SOURCES with KID that matches with CONTAINED_IN_KID. The listing should contain the authors, titles, publication, publisher, pages, and plates. This functionality is already implemented in the procedure for data presentation of the online database for a given unit identifier or unit name. This should be also implemented for the interactive link data.

3) If the query records in UNITS table with CURRENT_UNIT_KID is not null, then the query should switch to records with KID matching the CURRENT_UNIT_KID value. The output list of citations must then contain the name of the unit (PARENT_USAGE) and the prior name (PARENT_USAGE) for all records with CURRENT_UNIT_KID equal to KID.

4) The Oracle database could contain further text descriptions as Memo fields and image data, which could be used for drawings or picture files.

4. Customizing the web page

As described in chapter 2, there is a standard format available for the web output of the geological maps. For these interactive maps, customizing may be desirable for the web service. This is done by editing the Java script and Java parameter files directly on the UNIX platform with the VI editor.

4. 1. Directories and data organization

Because of the use of available data there is a good reason to define the data directories for the project for each element of the project. The directories depend on standards of the web service, the implementation of the Apache webserver, the ArcIMS system and the common data directories at KGS.
For the pilot project on Interactive maps for Montgomery County, several data and system directories were created, which are listed below. For further developments, it is recommended that this directory structure be maintained and other subdirectories added following the entire description.

AXL files

/home/crude2_2/imsmgr/AXL

Web pages

http://www.kgs.ukans.edu/arcims/montgomery_imap

Map Services

/home/crude2_2/arcIMS/maps

Images

/home/crude2_2/arcIMS/arcims/montgomery_imap/images

Arc Data

/home/crude2/acgina/aims/montgomery

Lexicon Data

http://polaris.kgs.ukans.edu/pls/abyss/autocarto.lexicon.lexicon

4. 2. Changes in Design of Interactive map

4. 2. 1. Top- Frame

The top frame contains a title of the web page and a logo. Both can be changed by editing the top.htm.

The title is defined in the head of the HTML page:

```
var theTitle = "IMAP Montgomery";
```

The image is defined as a reference in the body of the HTML page:

```
<IMG  SRC="http://polaris.kgs.ukans.edu/gifs/head/smKGShome2.gif"
WIDTH=33 HEIGHT=30 HSPACE=5 VSPACE=0 BORDER=0 ALT="KGS"
ALIGN="left">
```

4. 2. 2. Tool Bar

Tool bar changes may be defined in the toolbar.htm and the Java script functions of the interactive map, depending on the defined map services. Further details see the online handbook for ArcIMS: 'Customizing ArcIMS-HTML Viewer', [7].

4. 3. Attribute fields on Website

The different layers of the interactive map contain objects with alphanumeric data in attribute tables. These tables are defined by the shape files from ArcInfo. To change the column names or to limit the displayed fields in the web output, two modifications of the Java parameter file ArcIMSparam are needed. This can be done directly by the VI editor of the UNIX system, logging on as IMSMGR.

4. 3. 1. Definition of field selection

Limitation of fields, shown as object attributes on the web, can be achieved by setting the variable SELECTFIELDS from #ALL# to the needed field names list. Using several layers with different attribute fields this could be better managed by the variable SWAPSELECTFIELDS, which should be set to 'TRUE'. In this case the array from SELFIELDLIST must contain the name of each field to be displayed, separated from the next by a blank. The numbers of the array are determined from the list of layers. The first layer has the number '0', the second the number '1' and so on. For further details see online handbook: 'Customizing ArcIMS-HTML Viewer', [7].
The definition of fields for the pilot project on Montgomery County is listed in the 'Project data'.

4. 3. 2. Definition of Alias Names

There is a good reason to define aliases instead of original field names for the table output. To define these names, the variable SETFIELDALIAS must be set to TRUE. Then the array FIELDALIASLIST must define the list of fields and aliases for each layer. The number of array elements is the same as the number of layers. The syntax of this list is shown in the online handbook: 'Customizing ArcIMS-HTML Viewer', [7] .

The definition of aliases for the Montgomery County project is listed in the next section 'Project data'.

4. 3. 3. Project data

Layer name	Layer ID	Fields	Aliases
County_lines	0	BORDERING_ #ID# #SHAPE	Border Identifier Type
Township	1	NOTATION #ID# #SHAPE	Grid notation Identifier Type
Highways	2	HIGHWAYS_I FEATURE_TY FEATURE_NA #ID# #SHAPE#	Number Type of Highway Name Identifier Type
Roads	3	ROADS_ID FEATURE_TY #ID# #SHAPE#	Number Type of road Identifier Type
Cities	4	AREALAND ANPSADPI POP100 #ID# #SHAPE#	Area Name Population Identifier Type
Lakes	5	AREA FEATURE_NA #ID# #SHAPE#	Area Name Identifier Type
Topography	6	HYPSO_ ALTITUDE #ID# #SHAPE#	Hypsography Altitude Identifier Type
Geolines	7	CONTA_NAME CERTAINTY #ID#	Geo contact Classification Identifier

		#SHAPE#	Type
Hydrology	8	FEATURE_TY FEATURE_NA #ID# #SHAPE#	Description Name Identifier Type
Section_lines	9	#ID# #SHAPE#	Identifier Type
Tectonic_lines	10	LENGTH FEATURE_TY FEATURE_NA #ID# #SHAPE#	Length Description Name Identifier Type
Occurrence	11	OCCUR-TYPE LONGITUDE LATITUDE REFERENCE_ WEB_GIF #ID# #SHAPE#	Description Longitude Latitude Geo reference Image Identifier Type
Geology	12	AREA STRAT_LEGE TYPE_REF PLSS DESCRIPT PLACE KID_2 STRAT_GIF #ID# #SHAPE#	Area Geological Unit Type classification Type location Description Place Stratigr.Lexicon Stratigr.Column Identifier Type

Definition of attribute fields and aliases for presentation as a web service of the Montgomery County interactive map

4. 4. Overview Map

An overview map can be defined in the Java script file ArcIMSparam.js. The overview map is independent of the layer in the interactive map service. If another map service is to be used, then the new map service with the needed layer must be defined before changing the Java script file.

For the project "Interactive Map Montgomery" another map service named "KS_STATE" was created by ArcIMS, to display the overview map.

There are two changes requiered in the Java script parameter file ArcIMSparam.js, which are described in the following sections.

4. 4. 1. URL of the map service

The interactive URL for IMS is:
var imsURL =
'http://www.kgs.ukans.edu/servlet/com.esri.esrimap.Esrimap?ServiceName=Montgomery_Imap';

The URL for the overview map is:
var imsOVURL =
'http://www.kgs.ukans.edu/servlet/com.esri.esrimap.Esrimap?ServiceName=KS_State';

4. 4. 2. Set the coordinates for new overview map

Because of the definition of the overview map in the Java parameter file, the coordinates of the initial and maximum extent of the map must set manually in the Java parameter file. After several tests the listed coordinates were found to be suitable.

Coordinates for Overview map:

```
//initial map extent
        var startLeft = -96.040000;
        var startRight = -95.410000;
        var startTop = 37.450000;
        var startBottom = 36.970000;
//maximum map extent
        var limitLeft = -101.620000;
        var limitRight = -94.020000;
        var limitTop = 40.160000;
        var limitBottom = 36.760000;
```

4. 5. AXL file

The AXL file contains the definition of the layers for the map service, the identifier for each of the layers and the objects inside the layers. This file can be created automatically by ArcIMS, but was changed manually because of several color definitions and text descriptions desired for the layer 'Geology'.

The 'EXACT value' gives the value of the 'Lookupfield'. In the project this field is the 'COLORID' from the attribute field of the shape file 'Geology' that presents the layer 'Geology'.

The 'label' contains the text which is presented in the legend of the interactive map.

The 'fillcolor' gives the value of the color definition in RGB values for the entire geologic unit. This is defined according to standards for geological maps.

The following list describes the changes for the layer 'Geology':

```
EXACT value="29"    label="Noxie Ss"                          fillcolor="139,86,255"
EXACT value="28"    label="Unnamed shale with Thayer coal bed"  fillcolor="153,0,153"
```

```
EXACT value="27"    label="Cottage Grove Ss"                          fillcolor="153,153,225"
EXACT value="26"    label="Iola Ls"                                   fillcolor="204,143,255"
EXACT value="25"    label="Lane Sh and Wyandotte Ls"                  fillcolor="204,195,255"
EXACT value="24"    label="PLattsburg Ls"                             fillcolor="218,104,0"
EXACT value="23"    label="Vilas Sh"                                  fillcolor="255,51,0
EXACT value="21"    label="Captain Creek Ls"                          fillcolor="153,51,0"
EXACT value="20"    label="Eudora Sh Mbr"                             fillcolor="255,104,0"
EXACT value="19"    label="Stoner Ls Mbr"                             fillcolor="255,159,0"
EXACT value="17"    label="Cheyenne Creek Ss bed"                     fillcolor="255,182,0"
EXACT value="16"    label="Hafer Run Sh"                              fillcolor="255,153,0"
EXACT value="15"    label="Onion Creek ss"                            fillcolor="255,102,51"
EXACT value="9"     label="Revard Ss bed"                             fillcolor="102,255,102"
EXACT value="8"     label="Upper Tallant Sh bed"                      fillcolor="204,255,51"
EXACT value="14"    label="South Bend Ls Mbr"                         fillcolor="255,153,102"
EXACT value="13"    label="Weston Sh"                                 fillcolor="0,102,51"
EXACT value="7"     label="Tonganoxie Ss Mbr and Iatan Ls Mbr"        fillcolor="204,255,153"
EXACT value="12"    label="Big Heart Ss bed"                          fillcolor="51,102,0"
EXACT value="6"     label="Vinland Sh Mbr and Westphalia Ls Mbr"       fillcolor="153,153,0"
EXACT value="11"    label="Undifferentiated Tallant"                  fillcolor="153,204,0"
EXACT value="5"     label="Haskell Ls Mbr"                            fillcolor="102,102,0"
EXACT value="10"    label="Middle Tallant Sh"                         fillcolor="51,153,0"
EXACT value="4"     label="Robbins Sh Mbr"                            fillcolor="204,204,0"
EXACT value="3"     label="Ireland Ss Mbr"                            fillcolor="153,153,0"
EXACT value="2"     label="Terrace Deposits"                          fillcolor="255,255,102"
EXACT value="45"    label="Altamont Ls"                               fillcolor="200,141,167"
EXACT value="1"     label="Alluvium"                                  fillcolor="255,255,204"
EXACT value="44"    label="Nowata Sh"                                 fillcolor="255,153,153"
EXACT value="43"    label="Lenapah Ls"                                fillcolor="158,0,0"
EXACT value="42"    label="Lost Branch Fm"                            fillcolor="0,153,204"
EXACT value="41"    label="Seminole Fm"                               fillcolor="255,0,0"
EXACT value="40"    label="Checkerboard Ls"                           fillcolor="204,0,204"
EXACT value="39"    label="Tacket Fm"                                 fillcolor="180,204,255"
EXACT value="38"    label="Bethany Falls Ls Mbr"                      fillcolor="110,71,255"
EXACT value="37"    label="Ladore Sh"                                 fillcolor="0,98,204"
EXACT value="36"    label="Mound Valley Ls"                           fillcolor="0,153,255"
EXACT value="35"    label="Galesburg Sh"                              fillcolor="0,204,204"
EXACT value="34"    label="Dennis Ls"                                 fillcolor="102,255,255"
EXACT value="33"    label="Cherryvale Fm"                             fillcolor="153,204,255"
EXACT value="32"    label="Drum Ls (Fm)"                              fillcolor="204,255,255"
EXACT value="31"    label="Nellie Bly Fm"                             fillcolor="0,102,153"
EXACT value="30"    label="Dewey Ls"                                  fillcolor="27,227,127"
```

5. Conclusions

As shown by the example of the Montgomery County geologic and topographic maps situation, ArcIMS can be used to produce a suitable geological map on the Internet offering several services for the user of the system. The attribute tables of the shape files, containing additional descriptions of the graphical objects, are presented on the Internet using the information icon of the interactive map.

With some customizing of the standard output of ArcIMS, hotlinks with URL and image file names can be defined as hyperlinks inside the attribute tables. In this project there are several images with stratigraphic columns integrated into interactive maps on the web. They are presented in a separate browser window when the user clicks on an area of any geological unit in the active layer.

To define the access to an online database and to start the selection procedure, some program development is required. This is mainly a procedure package on Oracle that reads a given parameter, selects the data from the geological name database and presents the data in a table on a different web page. To connect this database to the interactive map, the parameter must be declared to trigger the select command in the database. This parameter is a value in the attribute field of the geological layer, which also contains the URL of the database access and the unit identifier of the desired geological unit. The hyperlink to this system then can be defined in the ArcIMS using the hotlink definition.

To integrate other geologic maps, this pilot project on Montgomery County could be reproduced to create additional interactive maps. The database procedures are ready for use in those applications without any changes.

It is shown in this pilot project that it is possible to define interactive maps with access to related archives using ArcIMS. The result integrates geologic maps and multimedia information from databases. The online map service opens the way to new dimensions in information for the web user, through multimodal and multimedial interaction based on hyperlinks.

6. References

[1] **Feuerstein, S., Pribyl, B. & Dawes, C. (1999)**: ORACLE PL/SQL Language.-).- O'Reilly, 94 p., ISBN 1-56592-457-6, Sebastopol, 1999

[2] **Feuerstein, S. & Pribyl, B. (1997)**: ORACLE PL/SQL Programing.- O'Reilly, 2nd. Edition, 987 p;, ISBN 1-56592-335-9, Sebastopol., 1997

[3] **Musciano, C. & Kennedy, B. (1997)**: HTML- The Definitive Guide.- 2 nd. Edition, 531 p., O'Reilly, ISBN 1-56592-235-2, USA 1997

[4] ORACLE Application Server- PL/SQL Web Toolkit.- 43 p., Part A60119-02, **Oracle Corp.** 1996, 1998

[5] **Baars, D.L. and Maples, C.G. (1998)**: Lexicon of Geologic Names of Kansas (through 1995).- Bulletin 231, Kansas Geol. Survey, 271 p., Lawrence, Kansas, 1998

[6] ORACLE Application Server- Developer's Guide: PL/SQL Applications. Release 4.0.8.2 - Part A66958-04, **Oracle Corp.**, June 2000

[7] http://arconline.esri.com, Handbook 'Customizing ArcIMS-HTML Viewer', **Environmental Systems Research Institute, Inc.**

[8] Using ArcIMS.- **Environmental Systems Research Institute, Inc.**, USA, 2000

[9] **J. A. Ross, D. R. Collins, D. Doherr (2001)**: Interactive Geological Maps using ArcIMS, with links to related data and images .- U.S. Geological Survey Open-File Report, 2001 (in press)

7. Appendix

AXL file

ArcIMSparam.js

Lexicon.sql (Sourcecode of the procedures for interactive map link)

AXL file Code

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<ARCXML version="1.0.1">
  <CONFIG>
    <MAP>
      <PROPERTIES>
        <ENVELOPE minx="-95.964485" miny="36.999050" maxx="-95.517662" maxy="37.386196"
              name="Initial_Extent" />
        <MAPUNITS units="DECIMAL_DEGREES" />
      </PROPERTIES>
      <WORKSPACES>
        <SHAPEWORKSPACE name="shp_ws-0" directory="/home/crude2/acgina/aims/montgomery" />
      </WORKSPACES>
      <LAYER type="featureclass" name="geology" visible="true" id="0">
        <DATASET name="geology" type="polygon" workspace="shp_ws-0" />
        <VALUEMAPRENDERER lookupfield="COLOR_ID">
          <EXACT value="29" label="Noxie Ss">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="139,86,255" filltype="solid"
                boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="28" label="Unnamed shale with Thayer coal bed">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="153,0,153" filltype="solid" boundarytype="solid"
                boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="27" label="Cottage Grove Ss">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="153,153,225" filltype="solid"
                boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="26" label="Iola Ls">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="204,143,255" filltype="solid"
                boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="25" label="Lane Sh and Wyandotte Ls">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="204,195,255" filltype="solid"
                boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="24" label="PLattsburg Ls">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="218,104,0" filltype="solid" boundarytype="solid"
                boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="23" label="Vilas Sh">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="255,51,0" filltype="solid" boundarytype="solid"
                boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="21" label="Captain Creek Ls">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="153,51,0" filltype="solid" boundarytype="solid"
                boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="20" label="Eudora Sh Mbr">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="255,104,0" filltype="solid" boundarytype="solid"
                boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
          <EXACT value="19" label="Stoner Ls Mbr">
            <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
                filltransparency="1.0" fillcolor="255,159,0" filltype="solid" boundarytype="solid"
                boundarywidth="1" boundarycolor="0,0,0" />
          </EXACT>
```

```
<EXACT value="17" label="Cheyenne Creek Ss bed">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,182,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="16" label="Hafer Run Sh">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,153,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="15" label="Onion Creek ss">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,102,51" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="9" label="Revard Ss bed">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="102,255,102" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="8" label="Upper Tallant Sh bed">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="204,255,51" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="14" label="South Bend Ls Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,153,102" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="13" label="Weston Sh">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="0,102,51" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="7" label="Tonganoxie Ss Mbr and Iatan Ls Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="204,255,153" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="12" label="Big Heart Ss bed">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="51,102,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="6" label="Vinland Sh Mbr and Westphalia Ls Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="153,153,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="11" label="Undifferentiated Tallant">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="153,204,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="5" label="Haskell Ls Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="102,102,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="10" label="Middle Tallant Sh">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="51,153,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="4" label="Robbins Sh Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="204,204,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
```

```
<EXACT value="3" label="Ireland Ss Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="153,153,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="2" label="Terrace Deposits">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,255,102" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="45" label="Altamont Ls">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="200,141,167" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="1" label="Alluvium">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,255,204" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="44" label="Nowata Sh">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,153,153" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="43" label="Lenapah Ls">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="158,0,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="42" label="Lost Branch Fm">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="0,153,204" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="41" label="Seminole Fm">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="255,0,0" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="40" label="Checkerboard Ls">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="204,0,204" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="39" label="Tacket Fm">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="180,204,255" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="38" label="Bethany Falls Ls Mbr">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="110,71,255" filltype="solid"
      boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="37" label="Ladore Sh">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="0,98,204" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="36" label="Mound Valley Ls">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="0,153,255" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
<EXACT value="35" label="Galesburg Sh">
  <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
      filltransparency="1.0" fillcolor="0,204,204" filltype="solid" boundarytype="solid"
      boundarywidth="1" boundarycolor="0,0,0" />
</EXACT>
```

```
    <EXACT value="34" label="Dennis Ls">
      <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
          filltransparency="1.0" fillcolor="102,255,255" filltype="solid"
          boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
    </EXACT>
    <EXACT value="33" label="Cherryvale Fm">
      <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
          filltransparency="1.0" fillcolor="153,204,255" filltype="solid"
          boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
    </EXACT>
    <EXACT value="32" label="Drum Ls (Fm)">
      <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
          filltransparency="1.0" fillcolor="204,255,255" filltype="solid"
          boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
    </EXACT>
    <EXACT value="31" label="Nellie Bly Fm">
      <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
          filltransparency="1.0" fillcolor="0,102,153" filltype="solid" boundarytype="solid"
          boundarywidth="1" boundarycolor="0,0,0" />
    </EXACT>
    <EXACT value="30" label="Dewey Ls">
      <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
          filltransparency="1.0" fillcolor="27,227,127" filltype="solid"
          boundarytype="solid" boundarywidth="1" boundarycolor="0,0,0" />
    </EXACT>
    <OTHER>
      <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0"
          filltransparency="1.0" fillcolor="51,0,204" filltype="solid" boundarytype="solid"
          boundarywidth="1" boundarycolor="0,0,0" />
    </OTHER>
  </VALUEMAPRENDERER>
```

```
</LAYER>
<LAYER type="featureclass" name="occurrence" visible="false" id="1">
  <DATASET name="occurrence" type="point" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLEMARKERSYMBOL color="255,0,255" width="6" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="tectonic_lines" visible="false" id="2">
  <DATASET name="tectonic_lines" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" color="227,127,27" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="section_lines" visible="false" id="3">
  <DATASET name="section_lines" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" color="127,27,127" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="hydrology" visible="false" id="4">
  <DATASET name="hydrology" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" captype="round"
        jointype="round" color="0,0,255" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="geolines" visible="false" id="5">
  <DATASET name="geolines" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" color="27,227,27" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="topography" visible="false" id="6">
  <DATASET name="topography" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" captype="round"
        jointype="round" color="192,192,192" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="lakes" visible="false" id="7">
  <DATASET name="lakes" type="polygon" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0" filltransparency="1.0"
        fillcolor="0,0,255" filltype="solid" boundarytype="solid" boundarywidth="1"
        boundarycaptype="round" boundaryjointype="round" boundarycolor="0,0,0" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="cities" visible="false" id="8">
  <DATASET name="cities" type="polygon" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0" filltransparency="1.0"
        fillcolor="255,175,175" filltype="solid" boundarytype="solid" boundarywidth="1"
        boundarycaptype="round" boundaryjointype="round" boundarycolor="0,0,0" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="roads" visible="false" id="9">
  <DATASET name="roads" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" color="227,227,227" />
  </SIMPLERENDERER>
</LAYER>
<LAYER type="featureclass" name="highways" visible="false" id="10">
  <DATASET name="highways" type="line" workspace="shp_ws-0" />
  <SIMPLERENDERER>
    <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" color="27,127,127" />
  </SIMPLERENDERER>
```

```
      </LAYER>
      <LAYER type="featureclass" name="township" visible="false" id="11">
        <DATASET name="township" type="line" workspace="shp_ws-0" />
        <SIMPLERENDERER>
          <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" captype="round"
              jointype="round" color="51,102,51" />
        </SIMPLERENDERER>
      </LAYER>
      <LAYER type="featureclass" name="county_lines" visible="false" id="12">
        <DATASET name="county_lines" type="line" workspace="shp_ws-0" />
        <SIMPLERENDERER>
          <SIMPLELINESYMBOL transparency="1.0" type="solid" width="1" captype="round"
              jointype="round" color="227,227,127" />
        </SIMPLERENDERER>
      </LAYER>
      <LAYER type="featureclass" name="stratigraphy" visible="false" id="13">
        <DATASET name="stratigraphy" type="polygon" workspace="shp_ws-0" />
        <SIMPLERENDERER>
          <SIMPLEPOLYGONSYMBOL fillinterval="6" boundarytransparency="1.0" filltransparency="1.0"
              fillcolor="200,141,160" filltype="solid" boundarytype="solid" boundarywidth="1"
              boundarycaptype="round" boundaryjointype="round" boundarycolor="0,0,0" />
        </SIMPLERENDERER>
      </LAYER>
    </MAP>
  </CONFIG>
</ARCXML>
```

ArcIMSparam.js

```
// javascript file with parameters specific to calling page
//*************************************************************************//*   Projekt
Interactive Maps and stratigraphic lexicon link              *//*  This file was modified by D.
Doherr, KGS, June 25, 2001                *
//*************************************************************************
////***********************************************************************
//*                    parameters File for HTML Template                    *
//*************************************************************************
// get machine name
var hostName = document.location.host;
// make URL for getting mapservice catalog
var catURL = "http://" + hostName + "/servlet/com.esri.esrimap.Esrimap?ServiceName=catalog";
// make prefix for URL
var serverURL  = "http://" + hostName + "/servlet/com.esri.esrimap.Esrimap?ServiceName=";
//*********************************************************************
//*                    parameters set by Designer                       *
//*********************************************************************
var imsURL =
'http://www.kgs.ukans.edu/servlet/com.esri.esrimap.Esrimap?ServiceName=Montgomery_Imap';
var imsOVURL =  'http://www.kgs.ukans.edu/servlet/com.esri.esrimap.Esrimap?ServiceName=KS_State';
var imsQueryURL = '';
var imsGeocodeURL = '';
// variables for setting component colors
var mapBackColor = '220,220,210';
var ovBoxColor = '#ff0000';
var ovBoxSize = 3;
var zoomBoxColor = '#ff0000';
// variables for using individual components
var hasOVMap = true;
var hasTOC = true;
var useModeFrame = true;
//initial map extent
//var startLeft = -95.964485;
//var startRight = -95.517662;
//var startTop = 37.386196;
//var startBottom = 36.99905;
//maximum map extent
//var limitLeft = -95.96448516845703;
//var limitRight = -95.51766204833984;
//var limitTop = 37.38654327392578;
//var limitBottom = 36.99905014038086;
//initial map extentvar startLeft = -96.040000;var startRight = -95.410000;var startTop =
37.450000;var startBottom = 36.970000;
//maximum map extent
var limitLeft = -101.620000;
var limitRight = -94.020000;
var limitTop = 40.160000;
var limitBottom = 36.760000;
//Web site path
var formFilePath = "/home/crude2_2/arcIMS/arcims/montgomery_imap"
var usePan=true;
var usePanNorth=false;
var usePanWest=false;
var usePanEast=false;
var usePanSouth=false;
var useZoomIn=true;
var useZoomOut=true;
var useFullExtent=true;
var useZoomActive=false;
var useZoomLast=false;
var useIdentify=true;
var useMeasure=false;
var useSetUnits=false;
var useSelect=false;
var useQuery=false;
var useFind=false;
var useGeocode=false;
var useStoredQuery=false;
var useClearSelect=false;
var usePrint=false;
var useBuffer=false;
var useExtract=false;
var MapUnits = "Degrees";
var ScaleBarUnits = "Miles";
// End of Designer set parameters
var useHyperLink=true;
//var useBufferShape=false; // not implemented
var hasToolBarOnLayer=false;
```

```
// file locations for servlet connector form creation
var headerFilePath = formFilePath + "/header.htm";
var footerFilePath = formFilePath + "/footer.htm";
/************************************
* Basic Map parameters
************************************/
// variables for map pixel offset from upper left corner of frame
// horizontal offset
var hspc = 0;
// vertical offset
var vspc = 0;
//panning factor for arrow buttons
var panFactor = 0.85;
//zoom factors for v.3
var zoomFactor = 2
// margin factor for zooming in on selected lines and polygons - based on feature width and
height. . . margin will be selectMargin * width or height
var selectMargin = 0.25;
// margin margin factor for zooming in on selected points - based on full extent. . . margin will
be selectPointMargin * fullWidth or fullHeight
var selectPointMargin = 0.025
// show the scale factor
var showScalePercent=false;
// display coords in status line
var showXYs=true;
//variables for MapDrawing
// North Arrow
var drawNorthArrow = false;
var NorthArrowType = "4";
var NorthArrowSize = "15";
var NorthArrowCoords = "20,30";
var NorthArrowAngle = "90";
// Scale Bar
var drawScaleBar = false;
// MapUnits=DEGREES,FEET,METERS
// can MapUnits be changed by user?
var setMapUnits=true;
// ScaleBarUnits=KILOMETERS,METERS,MILES,FEET
var ScaleBarBackground = "TRUE";
var ScaleBarBackColor = "0,0,0";
var ScaleBarFontColor = "255,255,255";
var ScaleBarColor = "192,192,192";
var ScaleBarFont = "Arial";
var ScaleBarStyle = "Bold";
var ScaleBarRound = "1";
var ScaleBarSize = "14";
var ScaleBarWidth = "7";
var ScaleBarPrecision = 2;
var numDecimals = ScaleBarPrecision;
// Copyright blurb
var drawCopyright = true;
var CopyrightFont = "Arial";
var CopyrightStyle = "Bold";
var CopyrightSize = "8";
var CopyrightCoords = "4,4";
var CopyrightColor = "0,0,0";
var CopyrightBackground = "False";
var CopyrightBGColor = "255,255,255";
var CopyrightGlow = "True";
var CopyrightGlowColor = "255,255,255";
var CopyrightText = "KGS, D.Doherr, June 2001";
// Mode on Map
var drawModeOnMap = false;
var modeRefreshMap = false;
var modeMapColor = "255,255,255";
var modeMapGlow = "128,0,255";
var ovImageVar;
var ovBorderWidth = 2;
var ovExtentBoxSize = 2;
// setup test for Nav 4.0
var isNav4 = false;
var isNav5up = false;
var isNav = false;
var isIE = false;
if (navigator.appName == "Netscape") {
        isNav =true;
        if (navigator.appVersion.search("5.0") == -1)
                isNav4 = true;
        else isNav5up = true;
        }
else
        isIE = true;
/************************************
```

```
* Extended Map parameters
************************************/
// variables for ovmap offset
var ovHspc = 0;
var ovVspc = 0;
// color for Main Map zoombox in html hex RGB format
//var zoomBoxColor = "#ff0000";
// index of initial active layer. . . if more than or equal to layer count top layer used
var ActiveLayerIndex=99;
// variables for using individual components
var useTextFrame=true;
// use external window for dialogs
var useExternalWindow=false;
// colors for tables
var textFrameBackColor="White";
var tableBackColor="White";
var textFrameTextColor="Black";
var textFrameLinkColor="Blue";
var textFrameFormColor="Gray";
// LayerList visible at service load
var showTOC=true;
// set layer visibility according to LayerList or by custom programming
var toggleVisible = true;
// set layer visibility of OVMap according to LayerList or by custom programming
// imsURL must equal imsOVMap - depends on one LayerList
var toggleOVVisible = false;
// will the LayerList show all layers, not just those available at current scale
var listAllLayers = true;
// Mode on floating layer
var drawFloatingMode = false;
var modeLayerOn = false;
var modeLayerColor = "Black";
var modeLayerShadowColor = "White";
var modeLayerFont = "Arial";
var modeLayerSize = "4";
// does the overview map a layer on top of map?...
var ovMapIsLayer=true;
var webParams = "";
```

```
if (parent.MapFrame!=null) {
        webParams = parent.document.location.search;
} else {
        webParams = document.location.search;
}
/*************************************
* Interactive Map parameters
*************************************/
// Click points - Measure/Shape Select/Shape Buffer
var clickMarkerColor="255,0,0";
var clickMarkerType="Circle";
var clickMarkerSize="6";
/*************************************
* Identify/Select/Query/Buffer parameters
*************************************/
// search tolerance in pixels around click
var pixelTolerance=2;
// color of selected features in decimal RGB format
var selectColor="255,255,0";
// color of highlighted feature in decimal RGB format
var highlightColor="255,0,0";
// level of transparency of selected and highlighted color
var transparentLevel = "0.5";
// zoom to selected feature if only one is returned?
var zoomToSingleSelect = true;
// fields to be returned in identify/selection/query request. . . #ALL#=all fields
var selectFields= "#ALL#";
//var selectFields= "#ID# #SHAPE#";
// swap out the list of returned fields?
//If true, a list must be defined in selFieldList[n] for each layer to update selectFields
var swapSelectFields=true;
// array for each layer's returned fields if swapSelectFields=true
var selFieldList = new Array();
// sample set for world - if not #ALL#, id and shape fields required. Separate with a space
selFieldList[1]="BORDERING_ #ID# #SHAPE#";selFieldList[2]="NOTATION #ID#
#SHAPE#";selFieldList[3]="HIGHWAYS_I FEATURE_TY FEATURE_NA #ID#
#SHAPE#";selFieldList[4]="ROADS_ID FEATURE_TY #ID# #SHAPE#";
selFieldList[5]="AREALAND ANPSADPI POP100 #ID# #SHAPE#";
selFieldList[6]="AREA FEATURE_NA #ID# #SHAPE#";
selFieldList[7]="HYPSO_ ALTITUDE #ID# #SHAPE#";
selFieldList[8]="CONTA_NAME CERTAINTY #ID# #SHAPE#";
selFieldList[9]="FEATURE_TY FEATURE_NA #ID# #SHAPE#";
selFieldList[10]="#ID# #SHAPE#";
selFieldList[11]="LENGTH FEATURE_TY FEATURE_NA #ID# #SHAPE#";
selFieldList[12]="OCCUR-TYPE LONGITUDE LATITUDE REFERENCE_ WEB_GIF #ID# #SHAPE#";
selFieldList[13]="AREA STRAT_LEGE TYPE_REF PLSS DESCRIPT PLACE STRAT_GIF #ID# #SHAPE#";
selFieldList[0]="STRAT_LEGE TYPE_REF PLSS DESCRIPT PLACE KID_2 #ID# #SHAPE#";
// use the field alias in the data display?
//If true, a list must be defined in fieldAliasList[n] for each layer defining aliases for those
fields needing them
var useFieldAlias=true;
// array for aliases for each layer's returned fields if useFieldAlias=true
var fieldAliasList  = new Array();
```

```
// sample set for world - fieldname:alias pairs separated by a bar (|)... if no aliases, use
empty string ("")
fieldAliasList[1]="BORDERING_:Border|#ID#:Identifier|#SHAPE#:Type";fieldAliasList[2]="NOTATION:Gr
id notation|#ID#:Identifier|#SHAPE#:Type";fieldAliasList[3]="HIGHWAYS_I:Number|FEATURE_TY:Type of
Highway|FEATURE_NA:Name|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[4]="ROADS_ID:Number|FEATURE_TY:Type of road|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[5]="AREALAND:Area|ANPSADPI:Name|POP100:Population|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[6]="AREA:Area|FEATURE_NA:Name|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[7]="HYPSO_:Hypsography|ALTITUDE:Altitude|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[8]="CONTA_NAME:Geo contact|CERTAINTY:Classification|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[9]="FEATURE_TY:Description|FEATURE_NA:Name|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[10]="#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[11]="LENGTH:Length|FEATURE_TY:Description|FEATURE_NA:Name|#ID#:Identifier|#SHAPE#:
Type";
fieldAliasList[12]="OCCUR-TYPE:Description|LONGITUDE:Longitude|LATITUDE:Latitude|RÉFERENCE_:Geo
reference|WEB_GIF:Image|#ID#:Identifier|#SHAPE#:Type";
fieldAliasList[13]="AREA:Area|STRAT_LEGE:Geological Unit|TYPE_REF:Type classification|PLSS:Type
location|DESCRIPT:Description|PLACE:Place|STRAT_GIF:Stratigr.Column|#ID#:Identifier|#SHAPE#:Type"
;
fieldAliasList[0]="STRAT_LEGE:Geological Unit|TYPE_REF:Type classification|PLSS:Type
location|DESCRIPT:Description|PLACE:Place|KID_2:Stratigr.Lexicon|#ID#:Identifier|#SHAPE#:Type";
// parameters for setting up hyperlinks in data display
var hyperLinkLayers = new Array();
// layers to have hyperlink
var hyperLinkFields = new Array();
// field in those layers to be used for hyperlink
hyperLinkLayers[0] = "stratigraphy";hyperLinkLayers[13] = "geology";hyperLinkLayers[12] =
"occurrence";
hyperLinkFields[12] = "WEB_GIF";
hyperLinkFields[13] = "STRAT_GIF";
hyperLinkFields[0] = "KID_2";
// will the returned data be displayed in text frame?
var showSelectedData=true;
// will the returned features be drawn?
var showSelectedFeatures=true;
// maximum number of features returned from query
var maxFeaturesReturned=25;
// number of data samples retrieved for query form
var numberDataSamples = 50;
/*************************************
* Legend parameters - aimsLegend.js
*************************************/
// legend map size
var legWidth=170;
var legHeight=300;
var legFont="Arial";
var legTitle="Map Legend";
/*************************************
* Options parameters - aimsOptions.js
*************************************/
// allowing user to set options
var allowOptions=false;
```

```
/*************************************
* ClassRender parameters - aimsClassRender.js
*************************************/
// parameters for custom class rendering... overrides default renderer
var ClassRenderLayer = new Array();  // layers to have custom renderers
var ClassRenderString = new Array(); // initial custom renderer XML string for the layers
ClassRenderLayer[0] = "Cities";
ClassRenderString[0] = "";
/*
ClassRenderString[0] ='<VALUEMAPRENDERER lookupfield="population">\n<RANGE LOWER="0"
UPPER="1000000">\n<SIMPLEMARKERSYMBOL color="255,0,255" type="circle" size="4" />\n</RANGE>';
ClassRenderString[0] = ClassRenderString[0] + '<RANGE LOWER="1000000"
UPPER="2500000">\n<SIMPLEMARKERSYMBOL color="255,0,255" type="circle" size="6" />\n</RANGE>';
ClassRenderString[0] = ClassRenderString[0] + '<RANGE LOWER="2500000"
UPPER="5000000">\n<SIMPLEMARKERSYMBOL color="255,0,255" type="circle" size="9" />\n</RANGE>';
ClassRenderString[0] = ClassRenderString[0] + '<RANGE LOWER="5000000"
UPPER="10000000">\n<SIMPLEMARKERSYMBOL color="255,0,255" type="circle" size="12" />\n</RANGE>';
ClassRenderString[0] = ClassRenderString[0] + '<RANGE LOWER="10000000"
UPPER="30000000">\n<SIMPLEMARKERSYMBOL color="255,0,255" type="circle" size="16"
/>\n</RANGE>\n</VALUEMAPRENDERER>';
*/
/*************************************
* Geocode parameters - aimsGeocode.js
*************************************/
// maximum geocode candidates returned - default = 20
var maxGeocodeCandidates=20;
// minimal acceptable geocode score for candidate
var minGeocodeScore=50;
var geocodePointColor = "255,0,0";
var geocodePointSize = "15";
var geocodeLabelSize = "12";
// custom functions needed for Reverse Geocoding
var useReverseGeocode = false;
// the starting point. . . it all starts here on loading
function checkParams() {
        appDir = getPath(document.location.pathname);
// global for overview map. . . change if not on same frame as Map
        ovImageVar = document.ovImage;
        debugOn = 0;
        if (parent.TextFrame==null) {
                useTextFrame = false;
                useExternalWindow=true;
        }
        if (!hasLayer("measureBox")) useMeasure=false;
        if ((!useMeasure) && (!drawScaleBar)) useSetUnits=false;
        if (ovImageVar==null) hasOVMap = false;
        if (parent.TOCFrame==null) hasTOC = false;
        if (parent.ModeFrame==null) useModeFrame = false;
        if (!document.layers)  {
                if (hasLayer("theTop")) document.all.theTop.style.cursor = "crosshair";
                if (hasOVMap) ovImageVar.style.cursor = "hand";
        }
        if (hasOVMap) {
// size of ov map image
                i2Width = ovImageVar.width;
                i2Height = ovImageVar.height;
```

```
// position of ov map
//ovMapLeft = iWidth - (i2Width + 6);
//ovMapTop = 2;
        }
        if (webParams!="") {
                //alert(webParams);
                getCommandLineParams(webParams);
        }
// if starting extents zero'd then flag to get start from mapservice
        if ((startLeft!=0) && (startRight!=0)) getStartingExtent=false;
// if limit extents zero'd then flag to get max from mapservice
        if ((limitLeft!=0) && (limitRight!=0)) {
                getLimitExtent=false;
                enforceFullExtent=true;
        }
        if (ovBoxColor=="") ovBoxColor = "255,0,0";
//ovBoxColor = convertHexToDec(ovBoxColor);
        if (aimsNavigationPresent) {
// Set up event capture for mouse movement
                if (isNav4) {
                        document.captureEvents(Event.MOUSEMOVE);
                        document.captureEvents(Event.MOUSEDOWN);
                        document.captureEvents(Event.MOUSEUP);
//document.captureEvents(Event.MOUSEOUT);
                }
                document.onmousemove = getMouse;
//document.onmousedown = chkMouseDown;
                document.onmousedown = mapTool;
                document.onmouseup = chkMouseUp;
//document.onmouseout = chkMouseOut;
        } else {
                usePan=false;
                usePanNorth=false;
                usePanWest=false;
                usePanEast=false;
                usePanSouth=false;
                useMeasure=false;
                useZoomIn=false;
                useZoomOut=false
//useFullExtent=false;
                useZoomActive=false;
```

```
//useZoomLast=false;
        }
        if (!aimsBufferPresent) {
                useBuffer=false;
        }
        if (!aimsQueryPresent) {
                aimsBufferPresent=false;
                useQuery=false;
                useFind=false;
                useBuffer=false;
                useStoredQuery=false;
        }
        if (!aimsSelectPresent) {
                aimsQueryPresent=false;
                aimsBufferPresent=false;
                useSelect=false;
                useQuery=false;
                useFind=false;
                useBuffer=false;
                useStoredQuery=false;
                useClearSelect=false;
        }
        if (!aimsIdentifyPresent) {
                aimsSelectPresent=false;
                aimsQueryPresent=false;
                aimsBufferPresent=false;
                canQuery=false;
                useIdentify=false;
                useSelect=false;
                useQuery=false;
                useFind=false;
                useBuffer=false;
                useStoredQuery=false;
                useHyperLink=false;
        }
        if (!aimsGeocodePresent) {
                useGeocode=false;
                useReverseGeocode=false;
        }
        if (!aimsPrintPresent) {
                usePrint=false;
        }
        if (!aimsOptionsPresent) {
                allowOptions=false;
        }
        if ((aimsXMLPresent) && (aimsMapPresent)) {
                if (aimsClickPresent) clickFunction("zoomin");
                if (parent.ToolFrame!=null) parent.ToolFrame.document.location="toolbar.htm";
                startMap();
        } else {
                alert("Unable to load one of the following Javascript
                        Libraries:\naimsCommon.js\naimsXML.js\naimsMap.js");
        }
}
```

Lexicon data (Source Code): Link procedures

```
--------------------------------------------------------------------------
/*
||            Stratigraphical lexicon data with PL/SQL
||             Package LEXICON
||             Created 04.19.2001, reviewed 05.15.01
||             by D. Doherr, KGS
*/
--------------------------------------------------------------------------
create or replace package LEXICON as
        Function SEL_OLD_FULL_LOCATION_FOR_UKID (PARA1 in varchar2) return varchar2;
        Procedure WEBHEADER;
        Procedure WEBFOOTER;
        Procedure SEL_UNIT_KID (p1 in varchar2, p2 in varchar2);
        Procedure PR_MAINDATA (text in varchar2, lex_text in varchar2);
        Procedure SHOW_SEQUENCE (SEQKID in varchar2);
        Procedure SHOW_DATAHEAD
                (S_NAME in varchar2,S_USAGE in varchar2,
                 S_PARENT_USAGE in varchar2,S_STAGE_NAME in varchar2,
                 S_SERIES_NAME in varchar2,S_SYSTEM_NAME in varchar2,
                 S_NAME_ORIGIN in varchar2,S_GEO_EXTENT_SURFACE in varchar2);
        Procedure SHOW_DATABODY
                (S_authors in varchar2, S_year in varchar2, S_title in varchar2,
                S_publication in varchar2, S_pages_plates_etc in varchar2,
                S_publisher in varchar2, S_abstract in varchar2,
                S_old_full_location in varchar2);
end LEXICON;
--------------------------------------------------------------------------
/
show errors;
--------------------------------------------------------------------------
/*
||      Definition of package body with procedures and functions
*/
--------------------------------------------------------------------------
create or replace package body LEXICON as
--------------------------------------------------------------------------
/*
||      Show the web page header with Bitmap of KGS and Title
*/
--------------------------------------------------------------------------
procedure WEBHEADER as
begin
        htp.headopen;
        htp.title ('KGS-- Kansas Geologic Names Database--');
        htp.headclose;
        htp.bodyopen ('#F0FFF0');
        htp.tableopen;
                htp.tablerowopen;
                        htp.tabledata;
                                htp.img('/gifs/head/smKGShome2.gif','Left','KGS Home
Page','','width=97 heigth=49');
                htp.tablerowclose;
         htp.tableclose;
         htp.centeropen;
                 htp.print ('_____________________ ');
                 htp.print ('<font size=+1><b>Stratigraphic Lexicon Data</b></font>');
                 htp.print ('_____________________ ');
        htp.centerclose;
        htp.para;
        htp.hr;
        htp.para;
end WEBHEADER;
```

```
--------------------------------------------------------------------------
/*
||      Show webpage footer with reference and email contact
*/
--------------------------------------------------------------------------
procedure WEBFOOTER as
begin
        htp.para;
        htp.hr;
        htp.tableopen ( 'width="100%"');
                htp.tablerowopen;
                        htp.tabledata ('<a href="http://www.kgs.ukans.edu/kgs.html">Kansas
Geological Survey</a>');
                        htp.tabledata ('Comments to :');
                        htp.tabledata ('<a href="mailto:ddoherr@kgs.ukans.edu">DDoherr</a>');
                htp.tablerowclose;
        htp.tableclose;
        htp.bodyclose;
        htp.htmlclose;
end WEBFOOTER;
--------------------------------------------------------------------------
/*
||    Print the entire information about stage, formation, series and
||    from the UNIT_KID, if data are found
*/
--------------------------------------------------------------------------
Procedure PR_MAINDATA
       (text in varchar2,
        lex_text in varchar2)
        is
begin
        if text is null or lex_text is null then
                NULL;
        else
                htp.print ('<tr><td>');
                htp.print (text);
                htp.print ('</td><td>');
                htp.print (lex_text);
                htp.print ('</td></tr>');
        end if;
end PR_MAINDATA;
--------------------------------------------------------------------------
/*
||    Procedure for getting the request from interactive maps with the
||      two parameters p1 (Sequence KID) and p2 (Map-Unit-KID) and
||      producing the output of description of rock formation, series...
||      and existing citations for that
*/
--------------------------------------------------------------------------
procedure SEL_UNIT_KID
        (p1 in varchar2,
         p2 in varchar2)
        is
        COUNTER integer;
        CCOUNTER integer;
        LOCAL_OLD_FULL varchar2(4000);
/*
||The cursor UNIT gives the entire UNIT_KID from the SEQUENCES_DETAIL table for the given
|| parameter values of Sequence kid and Map unit kid.
*/
        cursor UNIT is
                select UNIT_KID
                from LEXICON.SEQUENCES_DETAILS
                where SEQUENCE_KID = p1
                and MAP_UNIT_KID = p2;
      UNIT_REC UNIT%Rowtype;
```

```
/*
||The cursor LEX gives all values for the given KID from cursor UNIT except OLD_FULL_LOCATION
from OCCURRENCES
*/
        cursor LEX is
        SELECT u.name, u.usage, u.parent_usage, u.system_name, u.series_name, u.stage_name,
                u.name_origin, u.geo_extent_surface, c.abstract, c.pages_plates_etc, s.authors,
                s.year, s.title, s.publication, s.publisher
                FROM LEXICON.units u, LEXICON.citations c, LEXICON.sources s                WHERE
u.kid = UNIT_REC.unit_kid
                AND u.kid=c.UNIT_KID
                AND c.source_kid = s.kid;
        LEX_REC LEX%rowtype;
begin
      WEBHEADER;
      COUNTER := 0;
      CCOUNTER :=0;
      open UNIT;
        loop
              fetch UNIT into UNIT_REC;
              exit when not UNIT%found;
                open LEX;
                        loop
                                fetch LEX into LEX_REC;
                                exit when not LEX%found;
                                        if CCOUNTER = 0 then
                                                htp.strong ('Query for UNIT KID ');
                                                htp.print (UNIT_REC.UNIT_KID);
                                                SHOW_DATAHEAD (LEX_REC.name, LEX_REC.usage,
                                                LEX_REC.parent_usage,  LEX_REC.stage_name,
                                                LEX_REC.series_name, LEX_REC.system_name,
                                                LEX_REC.name_origin, LEX_REC.geo_extent_surface)/*
||Between the datahead and the databody should be presented the content of the possibly multiple
||existing field OLD_FULL_LOCATION in OCCURENCES. To behoof of this the UKID is to be found for
||the entire entity and is to be presented on the monitor
*/
                                                LOCAL_OLD_FULL := SEL_OLD_FULL_LOCATION_FOR_UKID
                                                        (UNIT_REC.UNIT_KID);
                                                if Length (LOCAL_OLD_FULL ) >2 then
                                htp.hr;
                                                        htp.print (LOCAL_OLD_FULL);
                                                        htp.br;
                                                        htp.hr;
                                                end if;
                                        end if;
/*
||Output of DATAHEAD for only the first time of looping, counted by CCOUNTER;||every time output
of the DATABODY
*/
                                SHOW_DATABODY(LEX_REC.authors, LEX_REC.year, LEX_REC.title,
                                        LEX_REC.publication, LEX_REC.pages_plates_etc,
                                        LEX_REC.publisher, LEX_REC.abstract, ' ');
                                CCOUNTER := CCOUNTER + 1;
                        end loop;
                close LEX;
                CCOUNTER := 0;
                COUNTER := COUNTER + 1;
```

```
/*
||      Restriction of possible output lines to 1000  with the COUNTER! */
              exit when COUNTER > 1000;
      end loop;
      close UNIT;
        if COUNTER >= 1000 then
                htp.hr;
                htp.print ('Data uncomplete because of large number of rows...');
        end if;
      WEBFOOTER;
exception
       when others
       then
       WEBFOOTER;
       NULL;
end SEL_UNIT_KID;
-------------------------------------------------------------------------/*||       Shows the
selected lexicon information for data head*/-----------------------------------------------------
-----------------Procedure SHOW_DATAHEAD     (S_NAME in varchar2, S_USAGE in varchar2,
S_PARENT_USAGE in varchar2,     S_STAGE_NAME in varchar2, S_SERIES_NAME in varchar2, S_SYSTEM_NAME
in varchar2,     S_NAME_ORIGIN in varchar2, S_GEO_EXTENT_SURFACE in varchar2)        isbegin
       htp.strong (' as ');   htp.strong(S_NAME);           htp.br;        htp.strong(S_USAGE);
       htp.br; htp.tableopen;         PR_MAINDATA('Parent Name', S_PARENT_USAGE);
       PR_MAINDATA('Stage', S_STAGE_NAME);          PR_MAINDATA('Series', S_SERIES_NAME);
       PR_MAINDATA('System', S_SYSTEM_NAME);        PR_MAINDATA('Name origin', S_NAME_ORIGIN);
       PR_MAINDATA('Surface Extent', S_GEO_EXTENT_SURFACE); htp.tableclose;end SHOW_DATAHEAD;
```

```
----------------------------------------------------------------------/*||       Shows the
selected lexicon information for data body*/-----------------------------------------------------
-----------------Procedure SHOW_DATABODY    (S_authors in varchar2, S_year in varchar2, S_title
in varchar2,    S_publication in varchar2, S_pages_plates_etc in varchar2, S_publisher in
varchar2,       S_abstract in varchar2, S_old_full_location in varchar2)    isbegin
       htp.blockquoteOpen;           htp.strong(S_authors); if (length (S_YEAR)>1) then
       htp.strong ('(');             htp.strong(S_year);           htp.strong ('):');     end if;
       htp.print(S_title);           htp.print ('.-');     if (length (S_PUBLICATION)> 1) then
       htp.print(S_publication);                  htp.print (',');       end if; if (length
(S_PAGES_PLATES_ETC)> 1) then               htp.print(S_pages_plates_etc);
       htp.print (',');       end if;       if (length (S_PUBLISHER)> 1) then
       htp.print(S_publisher);              htp.Br; end if; if (length (S_ABSTRACT)> 1) then
       htp.print(S_abstract);       htp.Br; end if;        if length(S_OLD_FULL_LOCATION) > 1
then          htp.print (S_old_full_location);          htp.Br;       end if;
htp.blockquoteClose;end SHOW_DATABODY;-----------------------------------------------------------
----------/*
||      Function for preselecting the value of the OLD_FUll_LOCATION and return
||      this as a string variable or a zero value string
*/
-------------------------------------------------------------------
Function SEL_OLD_FULL_LOCATION_FOR_UKID
       (PARA1 in varchar2) return varchar2
       is
       SELECT_OLDLOC varchar2(4000);
begin
       SELECT o.old_full_location into SELECT_OLDLOC
              from LEXICON.units u, LEXICON.occurences o
              WHERE u.kid = PARA1
              AND u.kid=o.UNIT_KID;
       return SELECT_OLDLOC;
exception
       when others then
       SELECT_OLDLOC := ' ';
       return SELECT_OLDLOC;
end SEL_OLD_FULL_LOCATION_FOR_UKID;
-------------------------------------------------------------------
end LEXICON;
-------------------------------------------------------------------
/
show errors;
```