Marine ALSF GIS

MarineGIS

Technical Specification
Ver 1.4: Final

Marine Environment Protection Research
MEPF 04/07
Marine ALSF Offshore Geographical Information System (Marine ALSF GIS)

ABP mer marine environmental research

GeoData Institute

Marine ALSF GIS

Technical Specification

Authors: Jason Sadler, Homme Zwaagstra

Date	Version	Editor	Description
04/04/05	1.1	jds	Report UC0786
13/04/05	1.2	cth	Edits and ABP mer comments
28/04/05	1.3	jds	Final draft edits
28/03/07	1.4	jds	Final

Contents

Marine ALSF GIS

Technical Specification

---

1 Introduction

1.1 Overview

The Marine ALSF GIS project (MarineGIS) will provide a website and online facility to store a live register of research studies structured as a spatial metadatabase and, where available, supply direct links to research outputs. The planned database and web-enabled Geographical Information System (webGIS) tool will facilitate the storage, searching and retrieval of information related to the marine aggregates sector. This document provides a preliminary technical overview of the underlying software architecture and associated tools and technologies to be used for development.

Although the underlying technologies outlined in the document are unlikely to change significantly, the final specification of aspects of the development will be determined in response to feedback received through ongoing consultation of the User Requirements Specification.

In summary, the Technical Specification revolves around a flexible, robust approach to data management. The core of this approach is the governance of marine project data via associated geospatial metadata, as applied through an established metadata content standard. The metadata framework will define the channels of access to research project information, stored research reports and will be enabled by XML technology. The use of XML will allow for a highly flexible and configurable interface to the metadata, which will be ideal for delivery over the Internet.

1.2 Application Structure

This document outlines the technical strategy for achieving the functionality demanded of the MarineGIS website in the User Requirements Consultation.

1.2.1 Data Organisation

A crucial aspect of the project is data organisation. As discussed in detail in the User Requirements specification, the system does not purport to provide a marine data warehouse, but a database of ALSF projects, metadata records, research reports and possible links to sources of live data, as illustrated in Figure 1.

Figure 1: Organisation of data within the MarineALSF database

The framework for organisation of MarineGIS data must encompass the range of report outputs outlined by the User Requirements specification, such as: geophysical, ecological and metocean surveys.

1.2.2 Component Structure

The overall component structure of the MarineGIS website, including the interactions between these components, is illustrated in Figure 2. The rest of this document discusses these major components in detail in Sections 2 (Web Interface), 3 (Software Architecture), 4 (Data) and 5 (Back-end Support). The interaction between these components is illustrated through flow charts similar to Figure 2 and, as above, the colours used relate to the component type.

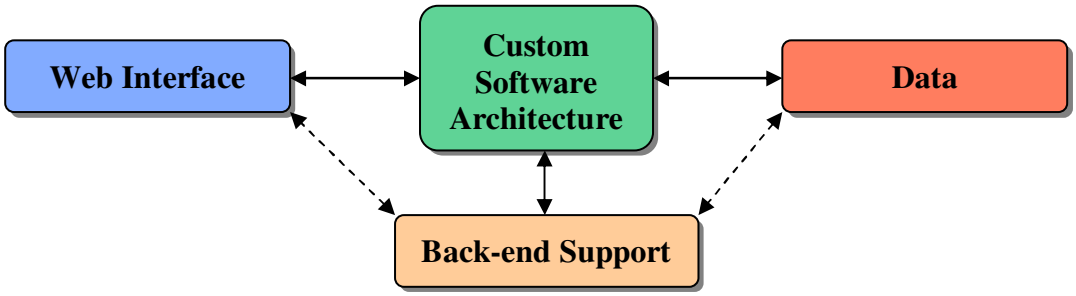

Figure 2: Interaction between the main components of the MarineGIS website

1.2.3 Metadatabase Architecture

A more specific illustration of the architecture used in the development of the spatially enabled Metadatabase is shown in Figure 3. As above, the specific components which will be utilized are described in more detail in later sections of the document.

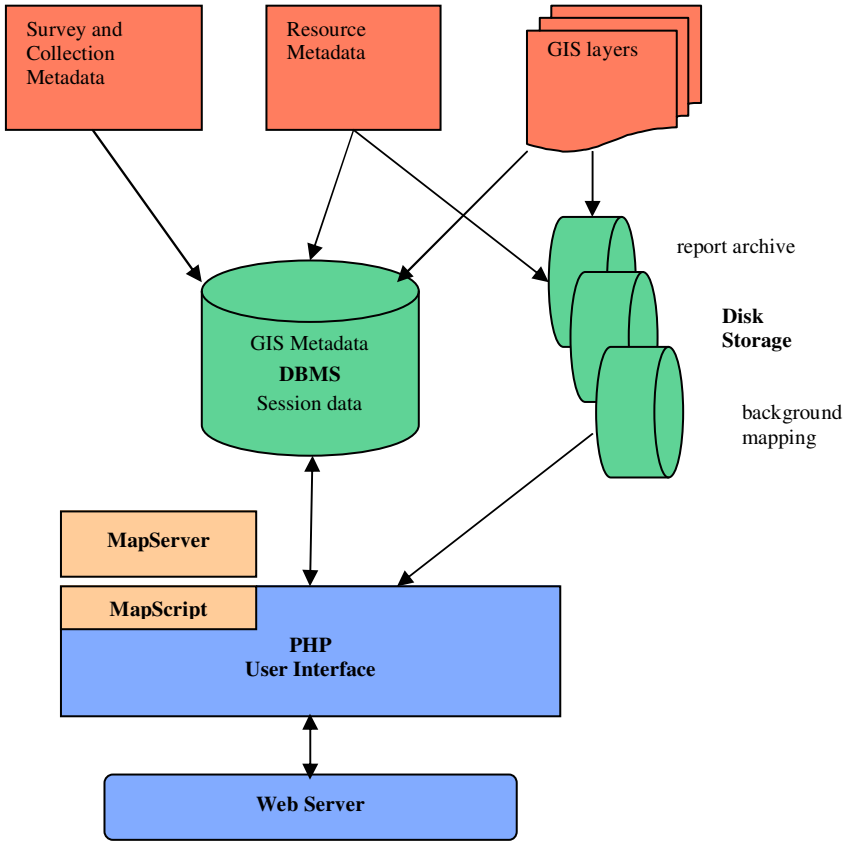

Figure 3: MarineGIS Metadatabase Architecture

2 Web Interface

The website front-end will make access to the back-end metadata as clean and intuitive as possible. The flow of logic involved in the creation of the web interface is described in Figure 4.

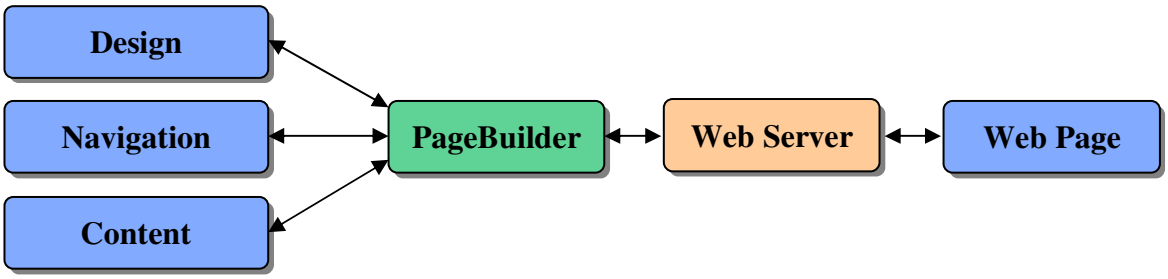

Figure 4: Process flow for MarineGIS web page creation

2.1 Design Template

A simple, sleek, contemporary design is proposed that will complement, but not detract from, the site content. Using the PageBuilder Content Management software developed in-house at GeoData, the design will be divorced from the content of the site, thereby simplifying updates and maintenance. Using this technique the functionality can be perfected through a frugal interface during development. Beautification of the interface can then become the focus prior to launching the site.

This template-driven approach to the site design, based entirely on templates and Cascading Style Sheets (CSS) also allows for the development of device specific templates for certain audiences (e.g. Wireless Applications Protocol (WAP) and text-only). Being a government-agency sponsored site, the recommendations detailed in the Guidelines for UK government websites[1] will be followed as far as possible. The final design will provide for a strong, site-wide identity and enable accessibility to a wide user-base.

2.2 Navigation

A clear segregation between the various functions of the MarineGIS website will define the site navigation. These areas would appear on standard navigation menus visible throughout the website:

- About MarineGIS
- Search
- Map Search
- Metadata
- Publications
- Events
- News
- Contacts
- Links

Being the core element of the site, access to the metadatabase will feature prominently in the navigation and design.

[1] Guidelines for UK government websites: http://www.cabinetoffice.gov.uk/e-government/resources/handbook/introduction.asp

2.3 Content

2.3.1 Static content

Appropriate textual and information-oriented imagery will be provided to flesh out the skeleton of the site described above.

2.3.2 Dynamic content

This will be generated from the Marine ALSF projects and associated metadata and research reports described in the User Requirements specification. Search and browse facilities based on metadata will be easily accessible and easy to use. For the most part these will be based on traditional web form selection techniques, but where the geospatial nature of archive data facilitates it (as defined by metadata); these forms will be complemented with map based interaction. Developed around MapServer scripting technology, this interface would be totally integrated within the main application.

3 Software Architecture

The majority of the application development will be concerned with the software architecture. This determines the channels through which users will interact with underlying archive data. The generalities of these interactions are described in Figure 5.

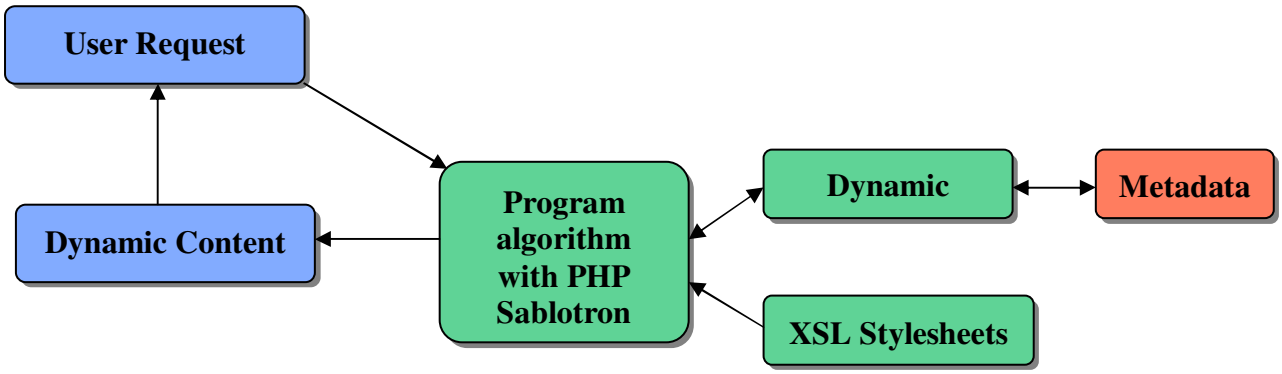

Figure 5: Process flow for archive data querying and processing

3.1 Component technologies

Before the details of the software architecture are discussed it will be helpful to introduce the core component technologies that may be integrated into the application.

3.1.1 Programming environment

The application will predominantly be written using the PHP scripting language[2]. PHP is a tool specifically designed for producing interactive web-based software and as such has excellent database support. Additional components may be written in C, Perl[3] or Python[4]. Use of these programming languages coupled with database abstraction will ensure portability across a broad spectrum of web hosting systems.

3.1.2 The Extensible Markup Language (XML)

XML is the universal format for structured documents and data on the Web[5]. It is envisaged that most project data and metadata will be manipulated by the program as XML. This is for reasons of data organisation and application inter/intra operability. A key technology for working with XML is the Extensible Stylesheet Language (XSL)[6]. A specific component of XSL called XSL Transformations (XSLT) will be exploited[7]. XSLT is a language for transforming XML documents into other XML documents.

To put this in context, the application will be delivered in the Extensible Hypertext Markup Language (XHTML)[8]. XHTML is a reformulation of the traditional HTML 4 as an XML application. This means that XSLT can be used to transform the native XML data into XHTML suitable for web use.

---

[2] PHP scripting language: http://www.php.net

[3] Perl scripting language: http://www.perl.com

[4] Python programming language: http://www.python.org

[5] Extensible Markup Language (XML): http://www.w3.org/XML

[6] Extensible Stylesheet Language (XSL): http://www.w3.org/Style/XSL

[7] XSL Transformations (XSLT): http://www.w3.org/TR/xslt

[8] The Extensible HyperText Markup Language (XHTML): http://www.w3.org/TR/xhtml1

The use of XML technologies and tools will enable the efficient generation of appropriate web interfaces directly from the adopted spatial metadata standard and the automatic validation of XML metadata records before they are stored in the database.

4 Web Map Service (WMS)

4.1 Introduction

The WMS is a recommendation produced by the Open GIS Consortium (OGC)[9] [10]. In essence, a WMS produces maps of georeferenced data for a requested area.

4.1.1 MapServer

The University of Minnesota MapServer is an Open Source[11] development environment for building spatially enabled Internet applications[12]. MapServer fully supports the IGC WMS specifications.

MapServer can accept both raster and vector input formats. Vector input formats include all those supported by the OGR library[13]. For performance reasons the ESRI Shapefile format is often used[14]. Raster input formats include all those supported by the Geospatial Data Abstraction Library (GDAL)[15].

The current version of MapServer supports output of 8 and 24 bit bitmapped images in PNG, GIF and JPEG formats as well as vector output in the form of shockwave flash, GML and PDF. Support for 24 bit bitmaps is significant because raster formats which support 24 bit colour such as ECW and TIFF do not have to be resampled down to 8 bit before they can be used.

4.1.2 MapScript

Through the MapScript[16] interface, the MapServer functionality is accessible to a variety of scripting languages including PHP, Perl and Python. This enables complete integration of the desired MapServer features into a custom application.

4.2 Program algorithms

This will be the core programming component of the application and as such will contain the logic to drive the functionality available to the user. Key functionality with regard to satisfying the XML demands of the application will be provided by the PHP Sablotron XSLT processor[17][18].

4.3 Application administration

Initially much of the application administration will be through the same low-level interface which facilitates the system administration of the UNIX platform on which the software will be developed. As the software architecture crystallises, however, administrative tasks can be transferred to a more portable web-based interface which would utilise the process flow described in Figure 5.

---

[9] Open GIS Consortium (OGC): http://www.opengis.org

[10] Web Map Service (WMS) implementation: http://www.opengeospatial.org/specs/

[11] Categories of Free and Non-Free Software: http://www.gnu.org/philosophy/categories.html

[12] MapServer: http://mapserver.gis.umn.edu

[13] OGR Simple Features Library: http://ogr.maptools.org/

[14] ESRI Shapefile Technical Description: http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf

[15] Geospatial Data Abstraction Library (GDAL): http://www.remotesensing.org/gdal

[16] MapScript: http://www.maptools.org/php_mapscript/index.phtml

[17] Sablotron: XSL Transformations Processor: http://www.xml.com/pub/r/231

[18] PHP's Newest Weapon: The Sablotron XSL Processor: http://www.xml.com/pub/r/895

4.4 Access

The MarineGIS database application will implement tiered levels of access to the metadatabase. This will support different types of user and define required access levels, such as Administrator, Moderator, Data Provider and Guest User. Even publicly available ALSF research reports may require a registration and login if a register of all downloads is to be provided.

5 Data

The underlying aim of the MarineGIS is the management of marine aggregate related project descriptions, metadata and research reports. As such, successfully achieving the aims of the project will be determined by how metadata are organised and managed, as this heavily influences such factors as application extensibility, ease of maintenance and ease of development.

5.1 Metadata structure

The metadata structure will be crucial to the organisation and management of the project information. The result of research for the User Requirements specification was a strong feeling that the UK GEMINI[19] metadata standard, as adopted by GI Gateway MetaGenie[20], would facilitate the necessary data description required by the project.

The details of this approach will have to be investigated with regard to the actual metadata required. This in turn is dependent on the uses to which the metadata will be put (i.e. Search ontologies). A further outcome of the research was an opinion that the UK GEMINI standard as applied to XML would be ideally suited to an online application. This was based around the capabilities of existing online metadatabase applications. Such a metadatabase could be wholly integrated into an application using the XML/XSLT tools available to the software architecture upon which the project will be built.

5.2 Data update procedure

A filter system based around quality assurance procedures is envisaged for metadata updates. In the first instance, ALSF project descriptions and metadata will be uploaded into the database structure by project partners. After this phase, it is envisaged that other appropriate parties would be able to upload project information, metadata and research reports. Uploaded data will then be quality assessed by staff with relevant access rights who will have the ability to edit metadata and make project information and outputs accessible to ordinary users.

5.3 Rate of growth of archive

Once established, the metadata structure would enable the growth of the ALSF database within that structure. Extending the structure to cope with new data types should be a straightforward exercise as the metadata structure and the implementation will be designed to be extensible enough to cope with future demands.

[19] UK GEMINI (Geo-spatial Metadata Interoperability Initiative): http://www.govtalk.gov.uk/schemasstandards/metadata_document.asp?docnum=903

[20] GI Gateway MetaGenie: http://www.gigateway.org.uk/metadata/metagenie.html

6 Back-end Support

This section deals with issues related to the support and hosting of the website. It is intended that the website be as independent of the particulars of these issues as possible.

6.1 Server Location

The application will initially be hosted on a GeoData Institute server within the University of Southampton for development and pilot purposes. The completed version could be hosted either at the same location or elsewhere, e.g. a preferred DEFRA service provider. On a technical level the website will be designed to allow deployment on alternative hosts after development. However, logistical considerations, such as restricting the installation of necessary software, may make porting the application unfeasible.

6.2 System Security

During development the website will be protected by firewalls. Within a firewall, website integrity is maintained by ensuring that the host operating system and application software are configured to an appropriately high level of security.

6.3 Backup Procedures

Onsite and offsite backup procedures will ensure data security. These procedures will be abstracted from the application, thereby removing a potential layer of complexity.

6.4 Operating System and Support Software

6.4.1 Operating System

The environment of choice for development will be a standard distribution of the Open Source Linux[21] operating system. Linux can also be recommended as a highly reliable and cost effective choice for the production system. However, the tools and technologies employed for developing the MarineGIS website and database are largely heterogeneous and can also operate in a Microsoft Windows or Commercial UNIX environment as an alternative to Linux, if required.

6.4.2 Database

A Database Management System (DBMS) will be required to store MarineGIS User, Session and Project data as well as Metadata and details of stored research reports. The DBMS must integrate with the development tools and be able to efficiently store and search geospatial information and XML data. The Open Source PostgreSQL database[22] with its geospatial PostGIS extensions[23] will be used for development. A high-end commercial alternative to these options could be the Oracle database[24] which can be spatially enabled through the Oracle Spatial and Locator options[25]. This option is not considered to be a cost-effective approach for the MarineGIS development.

---

[21] GNU Linux operating system: http://www.linux.org

[22] PostgreSQL database: http://www.postgresql.org/

[23] PostgreSQL PostGIS geospatial extension: http://postgis.refractions.net

[24] Oracle Database: http://www.oracle.com/database

[25] Oracle Spatial: http://www.oracle.com/technology/products/spatial/index.html

6.4.3 Web server

The web server used to deliver the MarineGIS application will be the Open Source Apache[26] HTTP server, representing the industry standard for this function. In a production environment, which may be influenced by service provider facilities, a commercial alternative to Apache or the Microsoft Internet Information Server (IIS) could also be deployed.

[26] Apache HTTP server: http://httpd.apache.org

7 Work Plan

The following is a list of the key tasks to push the development of the project forward from inception to implementation. They are listed in chronological order although some tasks will run concurrently.

1. Obtain feedback from this User Requirements and Technical specifications.
2. Make key decisions based on feedback e.g. User types, background data sets
3. Obtain a sample set of all types of data to be stored for testing purposes
4. Define specifics of core program algorithm including XML data storage techniques
5. Set up back-end infrastructure for the website
6. Implement functional design template
7. Code the application to implement the desired algorithms.
8. Develop the user interface to the program.
9. Test the application.
10. Enhance the application until desired functionality is achieved. This will be achieved through a cycle of repeating steps 7 – 9 in light of feedback from project partners.
11. Improve the design template.
12. Decide on final hosting and back-end support details.
13. Install first release of the application.
14. Test the application.
15. Public launch of the MarineGIS website

Defra is responsible for the Aggregate Levy Sustainability Fund (ALSF) and appointed CEFAS to distribute the Marine Environment Protection Fund (MEPF) element of the ALSF.

The ALSF has three main objectives:

1. **Minimising the demand for primary aggregates.**
2. **Promoting environmentally friendly extraction and transport.**
3. **Reducing the effect of local aggregate extraction.**

Information about the ALSF MEPF programme can be found on the website http://www.alsf-mepf.org.uk/default.htm

Further information about the Marine GIS is available on the website http://www.marinealsf.org.uk

CEFAS
The Centre for Environment, Fisheries & Aquaculture Science

The ALSF MEPF programme is administered by CEFAS
http://www.cefas.org.uk

defra
Department for Environment Food and Rural Affairs

Defra is responsible for the Aggregate Levy Sustainable Fund
http://www.defra.gov.uk/environment/waste/aggregates/