Marine Mammal Health Monitoring & Analysis Platform

System Design, Architecture and Schedule of Implementation

May 2018

System Design and Architecture: Marine Mammal
Health Monitoring and Analysis Platform (Health MAP)

2018

Felimon Gayanilo
Systems Architect
Gulf of Mexico Coastal Ocean Observing System (GCOOS)

Harte Research Institute for Gulf of Mexico Studies
Texas A&M University Corpus Christi, 6300 Ocean Drive, Unit 5869,
Corpus Christi, Texas 78412

Bibliographic Citation

Gayanilo, Felimon. 2018. System design and architecture: Marine mammal health monitoring and analysis platform (Health MAP), Harte Research Institute for Gulf of Mexico Studies, Texas A&M University Corpus Christi, 241p.

Persistent URL: http://data.gcoos.org/mmhmap/MMHMAP_SDA_v1.0.1.pdf
Language: English

Documentation Control

Version	*Description*	*Date*	*Remarks*
1.0.0	Initial version release	25 February 2018	Incorporated inputs from all sectors from a series of draft releases.
1.0.1	Updates from Samantha Simmons	08 May 2018	Additional edits and reformatting for consistency

About this Document

This document, System Design and Architecture: Marine Mammal Health Monitoring and Analysis Platform (Health MAP), was made possible through a funding support from the Marine Mammal Commission (the Commission). Special acknowledgment for the unwavering support and active participation in various activities leading to a robust design, to members of the Health MAP participants, led by Samantha Simmons of the Commission. Axiom Data Science, tasked with prototyping a visualization module to capture the essential functional features, has contributed in identifying technical requirements when browsing data. Special thanks to Mr. Naveen Kumar Kolli, Graduate Research Assistant, Department of Computing Sciences, College of Science and Engineering, Texas A&M University Corpus Christi, for his assistance with some sections of this report.

This document, System Design, Architecture and Implementation Schedules (SDA), is presented in several sections. This includes advanced sections that list the architectural components for data security and risk mitigation, vocabularies and community standards, organizational support structure, and document maintenance. The "Task and Module Implementation" section includes a breakdown of the various components of the design to allow for independent development of the components.

Extensive literature searches and interviews were conducted to derive the most appropriate technology stacks and guiding principles to meet the goals set for Health MAP. A presentation of the full design was made in May 2017 to solicit reactions to the proposed functional features. Efforts were made to make the SDA less technical, while remaining a technical reference when implementing the solutions. A copy of this document and all supporting documents are available online (http://data.gcoos.org/mmhmap).

Given the changing functional requirements and needs of the stakeholders, the SDA should be treated as a living document and subject to change without notice as the discussions on the elements of this design continue. A section on "Change Management" was added towards the end of this design document that addresses the recommended process to maintain the integrity of the information system. In addition, the SDA can serve as guidance to the Commission for budgeting, subcontracting, deployment of required modules, and implementing the governance to manage the lifecycle of implementing the recommended solutions.

Contents

Acronyms

ANSI	American National Standards Institute
API	Application Programming Interface
DB	Database
DBA	Database Administrator
DBMS	Database Management System
DIVER	Data Integration, Visualization, Exploration, and Reporting
DNS	Domain Name System
DR	Disaster/Recovery
DRG	Data Report Generator
DWC	Darwin Core
EML	Ecological Metadata Language
ETL	Extract, Transform, Load
FTP	File Transfer Protocol
GCOOS	Gulf of Mexico Coastal Ocean Observing System
GCOOS-RA	Gulf of Mexico Coastal Ocean Observing Regional Association
GUI	Graphic User Interface
HTTP	Hypertext Transfer Protocol
IOOS	Integrated Ocean Observing Systems
IP	Internet Protocol
IRI	Internationalized Resource Identifier
ISO	International Standards Organization
MMHSRP	Marine Mammal Health and Stranding Response Program
MMPA	Marine Mammal Protection Act
M2M	Machine-to-machine communication
NOAA	National Oceanic and Atmospheric Administration
NSDB	National Stranding Database
OP	OpenID Provider
RDF	Resource Description Framework
REST or RESTful	Representational State Transfer
RP	Relying Party
SDA	System Design, Architecture and Implementation Schedule
SQL	Structured Query Language
SWEng	Software Engineer
TAMU	Texas A&M University
TAMUCC	Texas A&M University Corpus Christi
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
WHISPers	Wildlife Health Information Sharing Partnership
WoRMS	World Registry of Marine Species
XML	eXtensible Markup Language
XSD	XML Schema Definition

Figures

Tables

Overview

The Marine Mammal Health Monitoring and Analysis Platform (Health MAP) program is a collaborative effort between the Marine Mammal Commission (the Commission), the National Oceanic and Atmospheric Administration (NOAA), National Marine Mammal Foundation and The Marine Mammal Center in support of the Marine Mammal Health and Stranding Response Program (MMHSRP) as outlined in Title IV of the Marine Mammal Protection Act (MMPA). The goal of Health MAP is to develop a comprehensive information system to collect, curate, and distribute data on marine mammal health that will give the public, scientists, and resource managers the ability to detect potential public and animal health risks, and to prioritize management and conservation efforts.

The preliminary list of functional features for the information system include:

- A platform for collecting, aggregating and publication of marine mammal health data;
- Enhance visual identification of "hot spots" of marine mammal health concerns;
- Web-based decision support tool(s) for resource managers;
- Resource for emergency responders; and
- Data portal to enable forecasting of marine mammal health for a given geographic area.

The current version of the information system, hereafter referred to as GulfMAP, aggregates marine mammal health data from stranding network members in the Gulf of Mexico. GulfMAP is the initial attempt to address the mission and vision of Health MAP. It defines the fundamental structure, i.e., elements, interfaces, processes, constraints and system functions of the information system that can work. The proposed system design and architecture documented here will be for an updated and scalable version of GulfMAP. The SDA should be treated as a living document and subject to change without notice. A section on "SDA Change Management" of this design document addresses the recommended process to maintain the integrity of the information system. Figure 1 provides a system overview as deduced from a series of meetings with the current committees and working groups focused on the development of Health MAP.

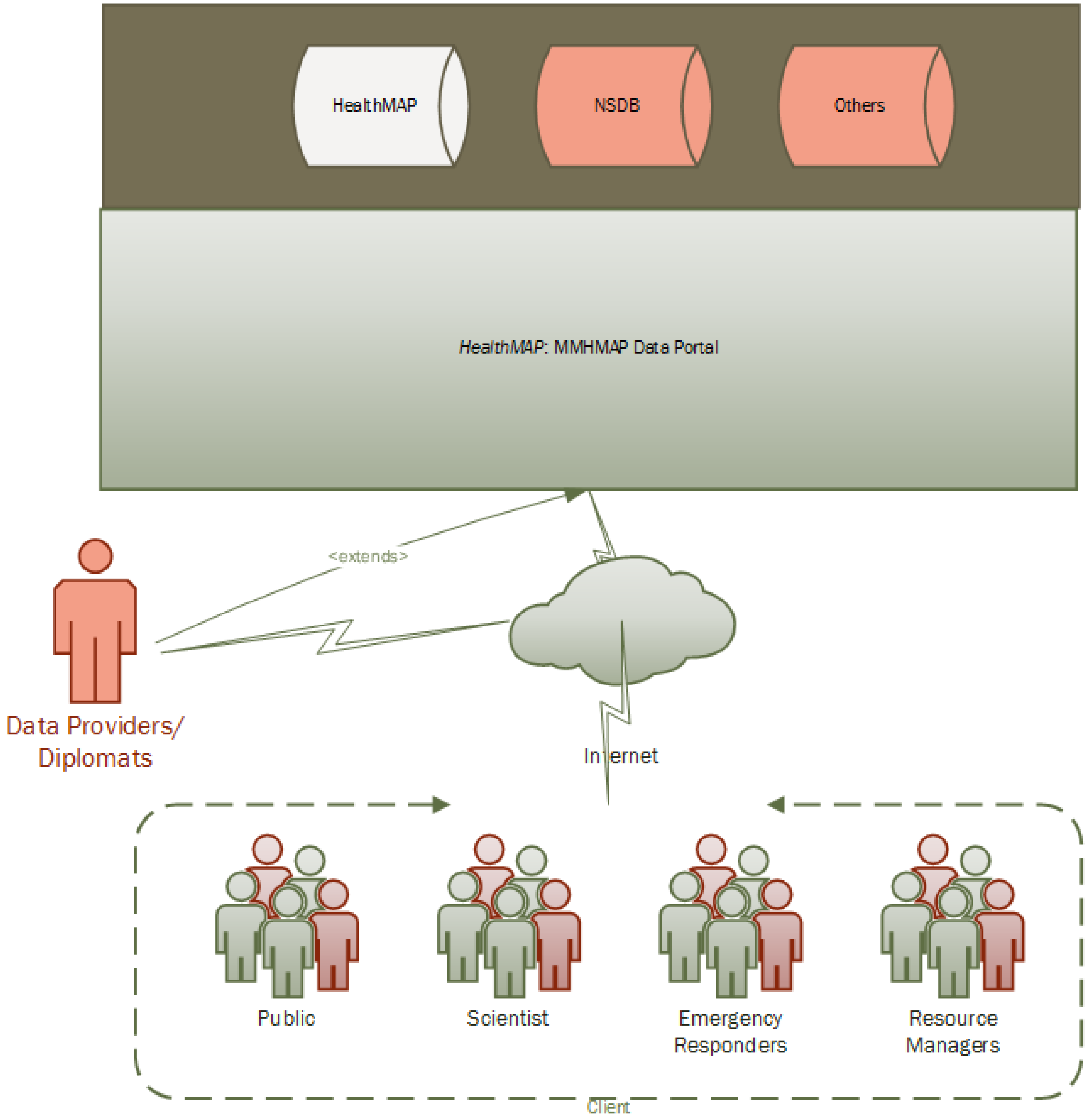


Figure 1. System overview of the Marine Mammal Health Monitoring and Analysis Platform (Health MAP)[1]

Health MAP should utilize state-of-the-art technology, like internet-based communications and data exchanges, and cater to a wide range of individuals, including scientists, emergency responders, resource managers, and members of the general public.

For the purposes of this design document, individuals with at least a high-school diploma and higher will constitute the term 'general public.' The term 'scientist,' will be defined as the scientists and researchers with common interest in the study of marine mammal health. "Resource managers' are similar to scientists but are focused on making science-based decisions in managing shared and renewable natural resources. Certified data providers are the individuals responsible for collecting field data, aggregating laboratory results that enrich the information system, and encoding the data for submission. The data diplomat trains the data providers on the system, audits data, and ensures that data are entered in a timely manner.

This design document identifies the elements of the information system (i.e., software, hardware, network, and personnel), communication and data exchange protocols, and procedures to assist NOAA and the Commission to realize Health MAP.

GulfMAP Use Cases and User Scenario

'Use Cases' were drafted based on interviews and a literature review to help define the existing elements and procedures, and the desired features and functional requirements to improve the information system. In many instances, defects and deficiencies of the current information system were revealed in this process. Notwithstanding the limitations of the present system as it is being developed in GulfMAP, examining use cases explains the processes involved in the collection, processing and archiving of marine mammal health data.

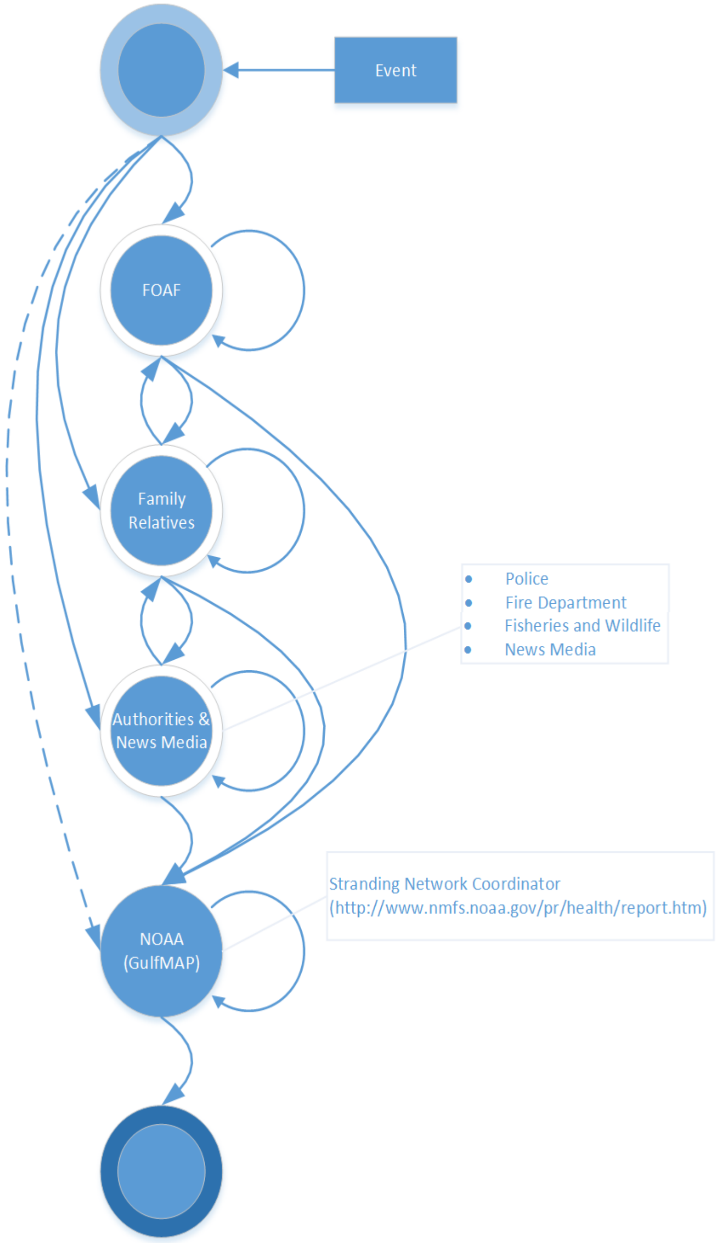


Figure 2. Incident notification flow. Information about an incident is sometimes circulated within a circle of friends (Friend-of-a-friend; FOAF) and within family member before it reaches authorities and news media.

Incident Notification

Reports of incidents, i.e., a stranding, entanglement or mortality event, originate from various sources. Figure 2 is a summary of how incidents reach NMFS and associates that trigger data collection. A report can arise from individuals and news spreads quickly to colleagues and the public. The event soon reaches the authorities (e.g., police, fire stations, NOAA) and news media that triggers a field visit by authorized data collectors. The period from when the incident was noted to the time a field visit is done varies considerably, from hours to weeks.

Although telephone numbers are listed on the NOAA Fisheries website (http://www.fisheries.noaa.gov/pr/health/report.htm) for all five of the regions (Alaska, Pacific Island, West Coast, Southeast/Carribean, and Northeast), these numbers are not widely known to the general public. Some regions have apps for use on mobile devices for marine mammal event notifications that expedite the process of reporting.

Field Data Recording

Data collectors are provided with a form for basic data ("Level A") collection and guidelines for to how to complete the Level A form (see Appendix A and B). Although the Level A form contains fields (spaces) to capture data in a standard format, data enumerators do write some additional data in available spaces, and even on the back of the form if available space is insufficient. Figure 3 illustrates the current data flow across all participating agencies.

Data are recorded in a local Microsoft Access database, hereafter referred to as the local DB, upon return from the field. This database is not web-enabled and only registered agencies (government or non-governmental) authorized by NOAA/NMFS have access to the standalone database. There is no online data bank that keeps a record of agencies or individuals participating in the program. Data are encoded directly onto the local DB and validated as they are populated. When a tissue sample is collected and sent to a recognized laboratory for further examination, the time needed for the lab results to return to the agencies varies

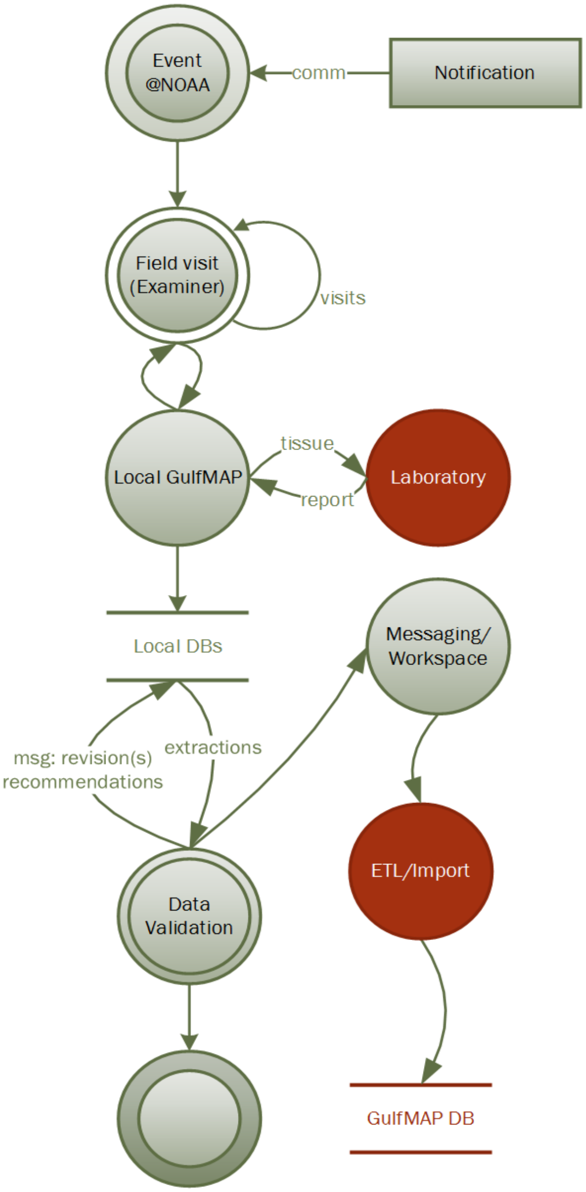


Figure 3. Basic data flow from the point of event recording.

greatly, from a day to a month. A copy of the local DB is uploaded to IOOS workspace monthly. The data on the local DBs are audited and imported, using an established ETL process (Extract, Transform, Load), onto a central GulfMAP based in the Southeastern Fisheries Science Center (SEFSC), Miami, FL.

User Authentication

GulfMAP is not an online system, and no user password management is needed. Data redaction is done internally to control data access. There are no set rules that govern the redaction process. Only data that were emailed from a recognized source are processed. A "Research Workspace" (https://researchworkspace.com) is, available, and it follows that same protocol, i.e., only folders from recognized individuals will be processed. There are no databases or registries maintained by recognized individuals and laboratories.

Disaster Recovery and Security

GulfMAP is a small database, and several copies of the database or database snapshots are made by the GulfMAP developer or database administrator that can be used to recover data. There are no "Disaster and Recovery Planning" documents and database maintenance, security, and recovery are relegated entirely to the database administrator. Several copies of the database exist and known only to the database administrator.

Report Requirements

GulfMAP and NOAA generate reports as the need arises. The only report produced with regularity is the Comma Separated Value (CSV) files needed to import data onto the National Stranding Database (NSDB). In some cases, NSDB also generates ad hoc reports for GulfMAP to ingest. Figure 4 is a summary of the reporting functionalities surrounding the application of GulfMAP.

In the planning phases, the same will be done to submit to or modify data at United States Geological Survey (USGS) Wildlife Health Information Sharing Partnership (WHISPers) [1] following a USGS designated template. A CSV file will be generated to facilitate ingestion of data onto WHISPers, from where the public can investigate the event.

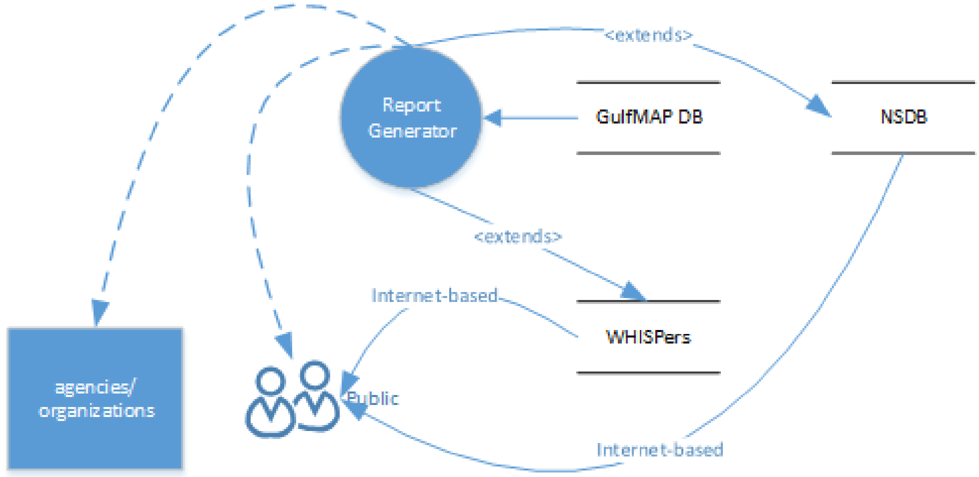


Figure 4. GulfMAP reporting requirements.

Data Browsing and Visualization

GulfMAP does not provide a graphic presentation of data entered. However, it has forms for data layout that users who have access to the database, can use to scroll through the holdings.

Organizational Structure

Marine Mammal Health MAP is currently led by a Steering Committee and supported by an Executive Committee. The Executive Committee includes individuals from the Marine Mammal Commission, NOAA Fisheries' (Marine Mammal and Sea Turtle Conservation Division), NOAA's National Ocean Services (Integrated Ocean Observing System or IOOS), and individuals from the private sector (National Marine Mammal Foundation and The Marine Mammal Center). Working Groups were formed to develop different components of Health MAP and regularly meet remotely via teleconference calls.

The technical development of GulfMAP is done largely by an individual from the Marine Mammal Health and Stranding Program of NOAA Fisheries, providing approximately, 30% of their time. Another full-time employee in the same office maintains the NSDB. Communication between these two individuals is regular and focuses primarily on data exported to the NSDB. Not all GulfMAP data are ingested to the NSDB given the absence of the necessary structure to absorb the data.

Health MAP Proposed Architecture

The proposed three-tiered architecture of Health MAP (Figure 5) is designed to meet the requirements identified by the Executive Committee and to scale from regional to national deployment as the need arises and/or as resources allow. It is composed of the user interface or 'presentation tier', application or 'domain logic tier', and 'the data storage tier'. An n-tiered architecture was selected to allow for a reuse of the components that will be developed for other purposes.

Data Tier

Health MAP will be utilizing internally built and developed databases, as well as externally managed databases in support of the operations and data processing. To facilitate scaling of the proposed information system, a relational database management system (DBMS) will allow for:

(i) multiple transactions to happen at any given time,

(ii) triggers to detect changes to the data,

(iii) views for consolidating tables,

(iv) conforms to ANSI/ISO/IEC 9075:2008 standard [2],

(v) open source system,

(vi) use of foreign keys for referential integrity, and

(vii) support on multiple platforms (e.g., Unix, MacOS, Windows).

A variety of DBMS, e.g., PostgreSQL [3] or MariaDB [4], have these attributes, but PostgreSQL and open source CentOS [5], a derivative of the Redhat Linux, are preferred based on the availability of experts.

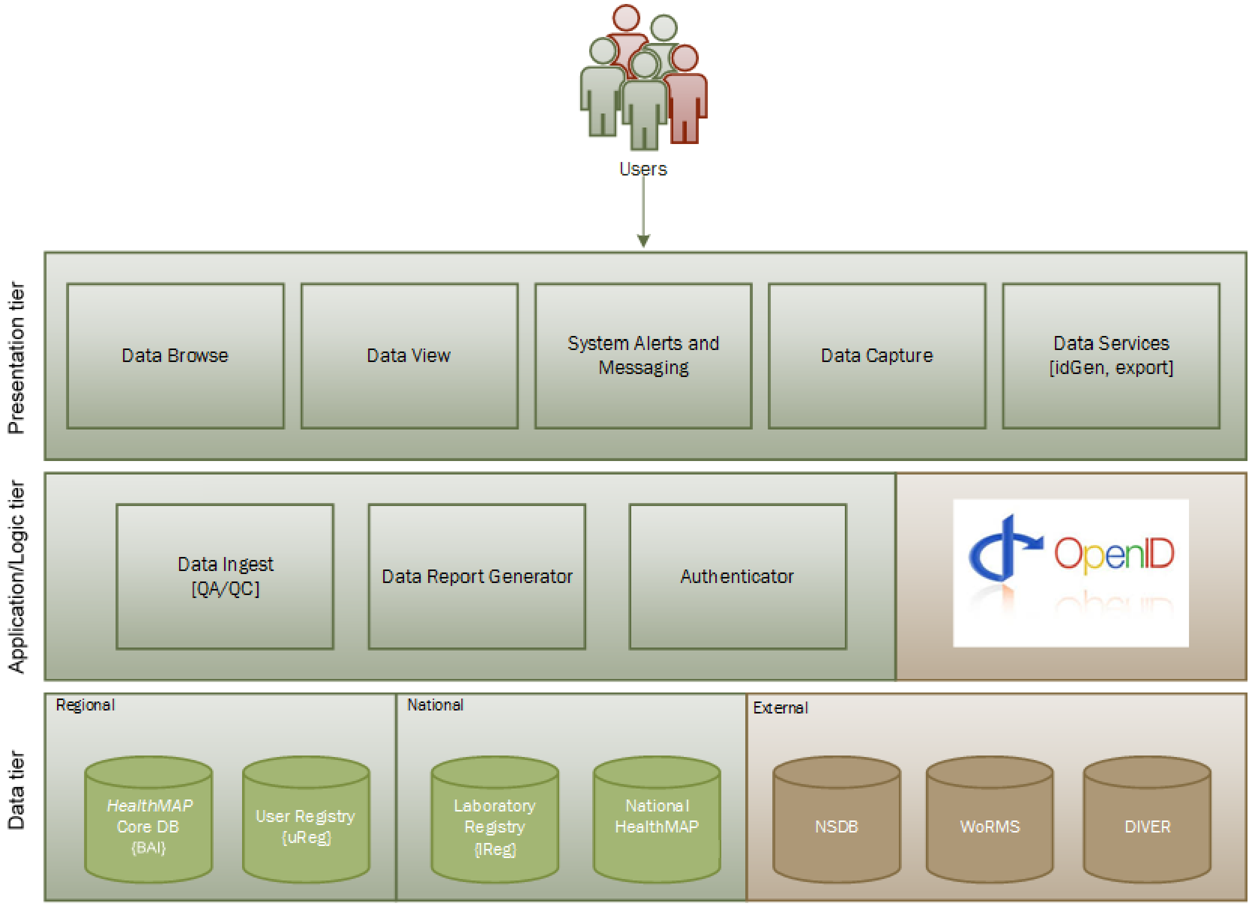


Figure 5. Health MAP multi-tiered architecture.

The primary dataset of Health MAP (referred to as, Basic Animal Information or BAI) will include tables and fields that are defined by the proposed data collection forms (Appendices A and B) and laboratory reports. The next section of this document, 'Data Capture and Encoding', describes the details of the data schema (Appendices B-G). Also, a User Registry database (referred to as uReg) is recommended to allow for efficient management of the users of the system. Theoretically, the list of users can be embedded in with the primary data set, but if the system components are to be reused for other purposes, a separate database will be most efficient and secures the user authentication details. The structure of uReg can be as simple as having three tables (Figure 6) to store the identifying elements of the user (core_uReg), audit trail (audit_uReg) to stamp why a change was made to the user and list of records the user encoded, and level of ownership. (Note this design document will not attempt to describe details of the table structures as this is a task associated with database design (see 'Tasks and Module Implementation' section) but it is presented here for clarification of essential function).

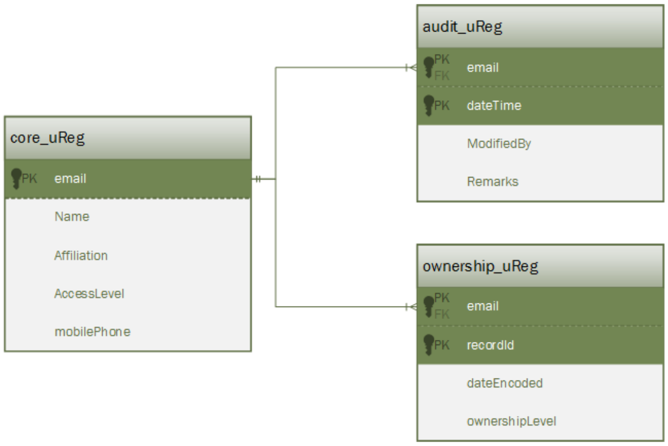


Figure 6. Database diagram of a simplified User Registry.

A registry of laboratories collaborating with NOAA (referred to as lReg) will also help in ensuring that facilities participating in the program comply with NOAA and Health MAP procedures. The proposed Laboratory Registry, unlike the User Registry, will be maintained at the national level. Figure 7 is a diagram of a possible lReg database that will allow listing the laboratories, their locations, contact details, and areas of expertise. Again, it is important to note that this design document will not attempt to get into the particulars of the table structure, as stated above.

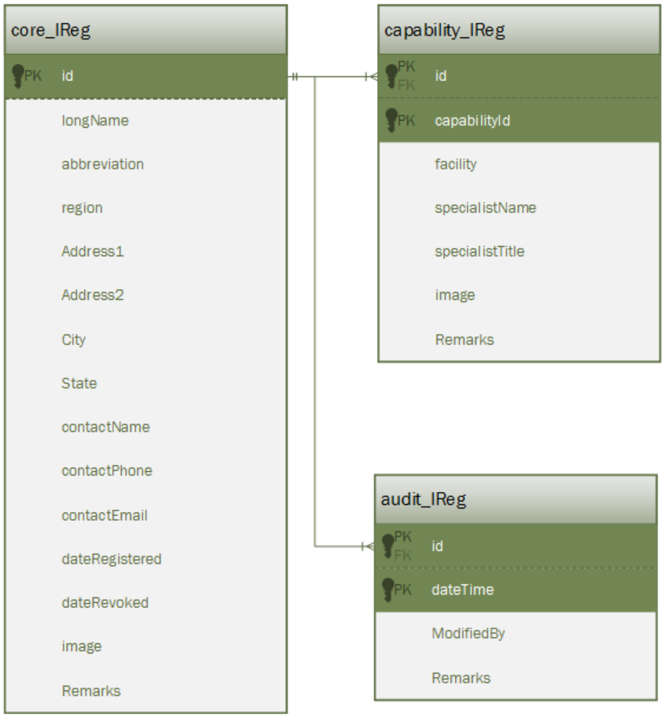


Figure 7. Data diagram of a possible *Laboratory Registry*.

There are several database engines available that can be employed for this information system, but it is recommended that only one should be used to simplify long-term maintenance. As the information system is scaled to a national level, i.e., presence of several regional systems, a national database management system will be required to serve as the point of reference and mirrored for all regional systems if a new regional database is then created. A separate procedure will be needed to synchronize independent records.

NOTE: Separate regional Health MAP databases are NOT recommended, but if a region warrants a distinct and independent information system, another instance can be installed and extended, as the region requires. However, changes to the data schema should remain compliant to the proposed data schema (Appendices C-G).

Other databases that will be linked to Health MAP once it is deployed include the National Stranding Database [11] (NSDB), the World Registry of Marine Species (WoRMS [12])for animal nomenclature and other details, the Wildlife Health Information Sharing Partnership - event reporting system (WHISPers [xx]) and NOAA's Data Integration, Visualization, Exploration, and Reporting system (*DIVER* [13]) which consolidates data from various sources. Communication between these external information systems will be managed by the application layer (see below).

Application/Logic Tier

Business rules on how to create, store and change data will be managed by this tier, Application/Logic Tier. There are three identifiable packages to interface the databases and the user interfaces: (1) Data Ingest, (2) Data Report Generator, and (3) Authenticator.

Data Ingest

'Data Ingest' is a core package that receives the data from user interfaces using the specified data schema, ensures data quality, saves the data, and when the messaging system is available, notifies the data provider and data administrator on the status of record ingestion. Data received that are not formatted using the proposed data schema (Appendices B to G) will be rejected. Table 1 summarizes these functional requirements for the Data Ingest package.

Table 1. Primary functional requirements of Data Ingest.

Priority[a]	Function	Remarks
*	Validate data using a predefined Schematron following the recommended data schema (Appendices C to G)	Software engineers should use Schematron whenever possible, but hard-coded rules can also be used.
*	Save captured and forwarded data by user interfaces onto BAI, uReg and lReg	Auditing tables should also be updated
1	Provide alerts and messages to appropriate users and auditing tables for changes on the databases.	Communications via emails and when appropriate, SMS for registered mobile devices.

[a] ' * ' are required functions, '1' for high priority and '2' for low priority.

Data Report Generator

Data Report Generator (DRG) is the opposite of Data Ingest. The DRG will extract the data from the repositories (regional, national and external databases) based on predefined SQL statements that will be determined by the report requirements. As a default function, the DRG will generate XML-based data following a data schema to contain the data or group of data requested (Appendices B to G; Figure 8).

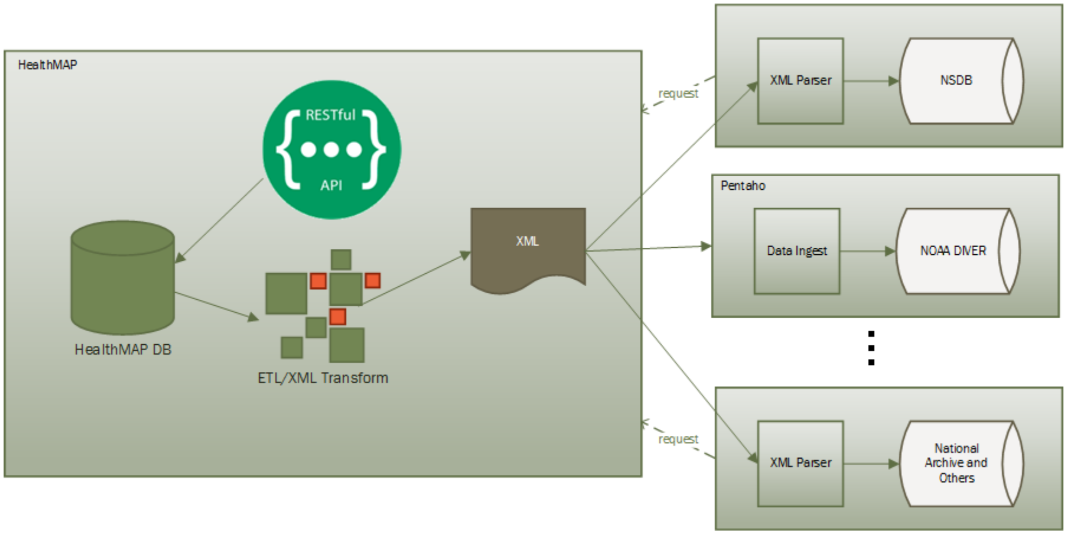


Figure 8 . Extract, Transform, Load (ETL) and XML generator as the primary data exchange strategy for Health MAP.

NOAA DIVER employs Pentaho Data Integration Platform [15] that can readily ingest XML formatted data such as what is proposed here. Other information system that requires inputs from Health MAP, such as the NSDB, will require the independent development of a parser to extract the data they will need to enrich their system.

NOTE: All Health MAP data converters/readers (i.e., XML parsers) specific for data ingestion on external databases will be maintained externally. However, it is highly recommended that software engineers who will develop the ETL/XML Transform module to create the first version of the XML Parser do so in close collaboration with the NSDB.

The DRG will contain a notification or messaging system to trigger responses or alert dependent elements of the information system (e.g., notify that the report is in the queue, ready, system encountered error, or report delivered and verified). The application of industry messaging technologies such as RabbitMQ [10], an open source message broker is highly recommended but not required. Messaging via a server-side text file will be sufficient as a trigger mechanism for this information system. Table 2 summarizes these functional requirements for the DRG.

Table 2. Primary functional requirements of the Data Report Generator.

Priority[a]	Function	Remarks
1	Provides a repository of SQL statements and allows the listing to be extended and managed (add, delete, edit)	A simple text-based file will suffice.
*	Generates an XML file based on SQL command and when applicable, follow the proposed data schema or extended version when required	When applicable, provide an option for reports to use Darwin Core
1	Provides alerts and status reports to appropriate modules.	Communications via emails and when appropriate, SMS for registered mobile devices.

[a] ' * ' are required functions, and '1' for high priority.

Authenticator

Authenticator is the package that checks/validates the credentials of the user and sets data access of users or systems. It is important to note here that Health MAP should not maintain user accounts (i.e., username and password). User authentication, i.e., user verification, should be done using user verification systems such as OpenID [6] or InCommon [7]. The underlying reasoning is simple, authentications should be managed (e.g., ensure good practice and security) to maintain the integrity of the system, and will require human resources to remain in compliance with federal mandates [14]. It is important to note here that, very often, users recycle their passwords. In many cases, users also use the same credentials used for financial and other applications that give access to confidential records. Monitoring authentication systems has become a challenge with increased attacks on cyber infrastructure, and it requires more human resources to manage authentication. Table 3 summarizes these primary functional requirements.

InCommon has matured in the last couple of years with the introduction of Shibboleth [8] and CILogon [9] utilities and support Application Programming Interfaces (API) to facilitate user verification through these technologies. However, its application remains limited, largely to academics and research institutions, because it requires membership to the network. Unlike InCommon, OpenID is a product supported by major technology companies like Google, Microsoft, Verizon, and PayPal, as well as security-conscious federal establishments such as the Office of the National Coordinator for Health Information Technology (http://HealthIT.gov). Google is the most common OpenID provider (OP), and proven to be more secure than most with the introduction of their 2-Step Verification system; hence, it is highly recommended here. Figure 9 is a sequence diagram that describes how OpenID is implemented.

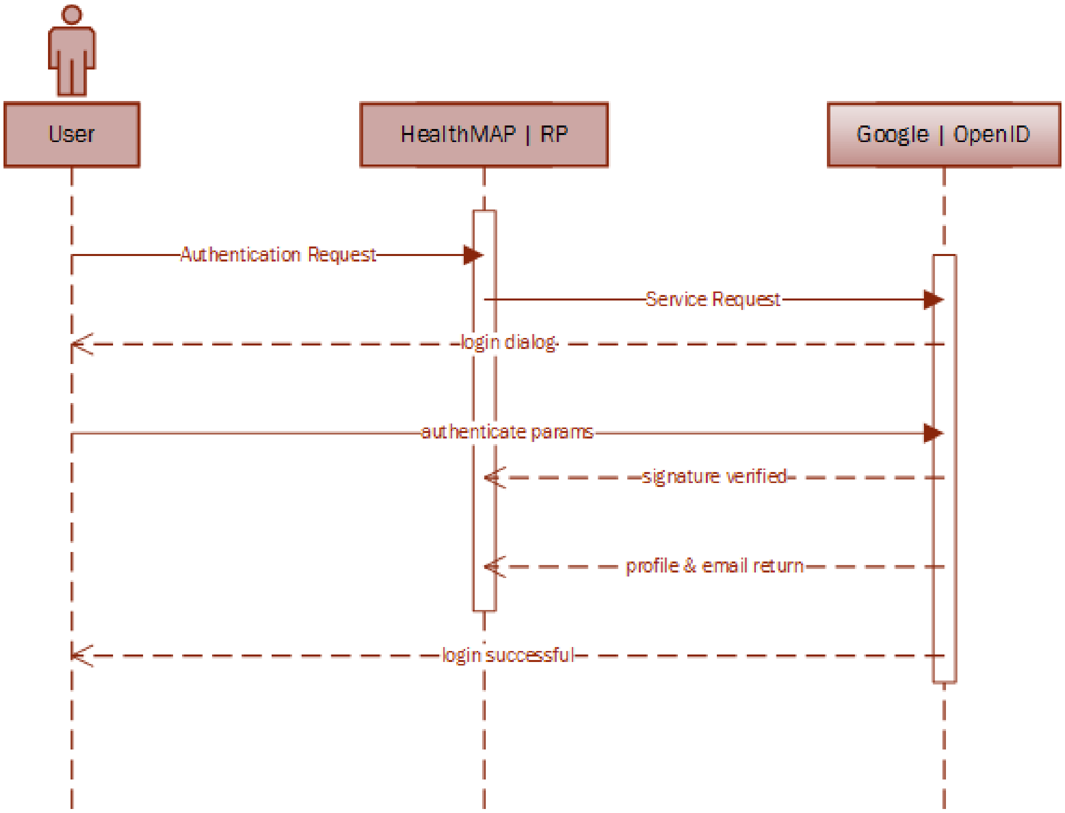


Figure 9. Sequence diagram when using OpenID.

Table 3. Primary functional requirements of the Authenticator.

Priority[a]	Function	Remarks
*	Communicates with an OpenID Provider and returns user profile and email.	Google is the recommended OP
*	Stores and retrieves the data access level of the user based on the uReg and starts a persistent session until the user logs out, the interface is closed or no actions are detected after an hour.	Setting or user interface to assign level of access is a function of the Data Services package
1	Updates the audit table to log user behavior within the system.	
1	Provides messaging functions that can alert administrators via registered mobile device or email	Communication option is configurable

[a] ‘ * ’ are required functions, and ‘1’ for high priority.

When a user logs in, Health MAP’s Authenticator (Relaying Party or RP) makes a service request to Google (here referred to as the OpenID Provider or OP) and Google presents a login form. After verification, Google sends to Health MAP Authenticator, the user profile and email that can be used to set the user data access level. The user credentials (username and password) are never relayed to Health MAP.

Working in collaboration with the Authenticator is the Data Services package in the Presentation Tier that will allow the system administrator to initialize the uReg to set data access controls. Appendix H is the current (April 2017) Health MAP Data Policy that enumerates the different levels of data access.

NOTE: The choice of the programming language or scripting environment is as important as the selection of the operating system. The commonly use scripting or programming languages that are most suited for a web-based application for the front-end and can equally be robust and flexible to handle the middle-tier are limited to PHP, Java, Python, Ruby, and Perl. This list is based on the following attributes: ease of use, exception handling, cross-platform portability, community acceptance and increasing popularity, availability of tools, performance, and security. The most important attribute for long-term maintenance is community acceptance. Python has a growing community and has proven to be a stable and matured scripting language that should be considered for the development of Health MAP.

Presentation Tier

The Presentation Tier or often what is referred to as the Graphical User Interfaces (GUI), are the modules that the public users will have access. The tier will include the Data Browse, Data View, System Alerts and Messaging, Data Capture and Data Services packages.

Data Browse

The Data Browse package is a user interface that will allow the users to explore the data via a search engine that can be configured by the user. The return results will be a list of records, in table form that the user can also set based on the content of the database. The fields and choice of records that can be viewed or configurable will be dependent on the level of data access provided to the user (refer to Health MAP Data Policy, Appendix H). Upon user verification (login via OpenID), additional fields can be configured and more records can be listed. Designated Health MAP System Administrators or those provided with 'Administrative' function, will also be able to view the contents of all audit tables in the system. Table 4 summarizes the primary functional requirements of the package.

Table 4. Primary functional requirements of the Data Browse package.

Priority[a]	Function	Remarks
*	Interface for a user-defined search of the database	Users should be provided with options to add or remove fields that they have access to.
2	Saves the user-defined SQL statements	the uReg should be modified to allow Health MAP to save these queries
*	Sends a request for data to DRG based on user request	RESTFul web service is preferred
*	Reformats the returned results in table format.	Users should be able to sort, provide a filter, to view the table results.
1	Updates the audit table to update the data extracted (as viewed data).	
1	Pass data and control to the Data View package to allow users to view the table results in graphical form.	

[a] ‘ * ’ are required functions, ‘1’ for high priority and ‘2’ for low priority.

Data View

Data View is similar to Data Browse but data are presented in a graphical format through a web browser, i.e., data are symbolized and overlaid on a georeferenced map. The choice of which fields can be viewed or grouped (aggregate function) should also be configurable as in the case of Data Browse (i.e., user interactive interface). Table 5 summarizes these primary functions. Plotting data points as a map or grouping them, is useful as it provides visual impact about the areas where the events occurred. Using different colors can also help in identifying 'hot' zones or quantifying the areas where the event occurs which helps in visually categorizing the data returned. An application that allows the user to readily change the viewport to a certain time-period (single date point or a range) provide users with the perspective on the trend of the data plotted. However, if these data points are contoured using any interpolation approaches, such as Kriging [22], the contoured map can reveal more information such as recruitment or sink areas or statistical likelihood of occurrences.

Table 5. Primary functional requirements of the Data View package.

Priority[a]	Function	Remarks
*	Web-based, user interactive map (terrain or satellite images) that allows for basic functions	The basic web-based interactive map includes panning, zoom-in and zoom-out, measure between two points, display latitude, longitude in decimal degrees, and feature labels.
*	Overlay user-defined data layers.	Group data as the need arises that is dependent on the density of data and zoom level.
1	User-configurable layer presentation	Presentation of the data points should not be limited to push single colored pins and hexagons, but users should be able to configure the symbols.

1	Download and print map view	Add data source and appropriate labels (year data collected, originator, etc.) to all report printout.
2	Contour mapping of data point	Although point maps are useful, Kriging the data points often reveals areas of interest.
2	Time-slider function to allow for an interactive view of the data layers for a given period using a slider widget	The time frame can be a range

[a] ‘ * ’ are required functions, ‘1’ for high priority and ‘2’ for low priority.

System Alerts and Messaging

The System Alerts and Messaging package will handle all messaging system required by Health MAP. The alerts and messages will be configurable by the user via an interactive web browser. These messages can be delivered to a registered phone, email or upon login. This package will include, but is not limited to the following configurable user-level alerts and messages (Table 6):

- System Status report (server communication state);
- Changes made to any of the records;
- Regular server site statistics or by demand (unique visitors, number of visits, pages, hits, and bandwidth);
- Data use metrics (number of data views, and downloads); and
- System administrator messages (reminders, emergency calls, and general short messages).

A third party monitoring system, such Uptime Robot [23] or Nagios [24], an open source infrastructure monitoring, can be employed to monitor the services independently. If an alternate site has been setup, a domain name rollover to Health MAP alternate site can be automated through the Domain Name System (DNS) Server, responsible for converting alphabetic names to Internet Protocol (IP) addresses of the domain. This approach, or redirecting domain names to another server if the primary server is down, is highly recommended (see section on Data Security and Risk Mitigation).

Table 6. Primary functional requirements of System Alerts and Messaging.

Priority[a]	Function	Remarks
*	Web-based user interface to configure alerts (on/off)	
1	Configurable alerts include server status (up or down), site statistics, changes to specific records or group of records, data view, and downloads.	Setting or user interface to assign level of access is a function of the Data Services package
1	Administrator alerts that send messages to all registered users	
*	System short message for all user interfaces	Posting of messages on user interfaces to users (registered or not), such as notifications of service interruptions.

[a] ' * ' are required functions, '1' for high priority and '2' for low priority.

Independent of the System Alerts and Messaging package, employment of a ListServe [16] or other mailing system is suggested to ensure regular communication and promote collaboration with stakeholders in Health MAP. This package should be configurable to allow users to subscribe or unsubscribe as the need arises.

Data Capture

Data Capture is the primary interface for data providers. A web-based system is recommended here to implement the data capture/collection form (see Appendix B). The popularity of mobile devices requires that the Data Capture be translated (re-coded) into a mobile App on iOS, operating system for Apple-based mobile devices, and Android, an operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. The primary advantage of mobile devices is that the location and photos of the animal or event can rapidly be entered into a database if the data enumerator is on site. The disadvantages of using mobile devices are: (1) the workspace is limited and designing the user interfaces can be challenging and (2) moving the data in the absence of WiFi, a wireless local area networking with devices based on the IEEE 802.11 standards, can be expensive.

NOTE: The working version of the GUI prototype based on recent recommendations of a Health MAP Working Group, can be downloaded from http://data.gcoos.org/mmhmap/Health MAP_Form.zip. This prototype works only in a Windows environment.

Data Services

The Data Services package is a collection of independent sub-modules for the following functions (Table 7):

- Generate Health MAP Record ID (see Data Capture and Encoding);
- Validate externally generated Record ID;
- Administer the User Registry (uREG) and Laboratories Registry (lREG) databases;

Table 7. Primary functional requirements of Data Services package.

Priority[a]	Function	Remarks
*	RESTful approach to generate or validate RecordID	see Persistent Record ID recommendations
*	Web-based user interface to manage users, and send data to Data Ingest for proper storage of data	see Appendix H for data access levels
*	Web-based user interface to manage the list of recognized laboratories, and send data to *Data Ingest* for proper storage of data	
2	Web-based interface to obtain metadata for the record where DOI is requested, and forwarded to DataCite for proper handling	Metadata following community standards such as ISO 19115-2 or EML
2	Interface to update a record or group of records for DOI stamping	
1	Request form to generate a statistical report of data downloads and views, send data to DRG and publish the report.	This report should include, among others, basic count statistics and group by period (e.g., monthly)
*	Web-based user interface to manage the drop-down list(s) for use with Data Capture package	

[a] ' * ' are required functions, '1' for high priority and '2' for low priority.

- Facilitate the registration of a record or group of records for Digital Object Identifier (DOI) [17] minting with DataCite [18] or other global DOI provider; and
- Statistical reports for data downloaded, viewed and other server statistics.

Server site statistics can be handled by off the shelve software packages such as AWStats [19], Piwik [20] or Google Analytics [21]. It should be noted that Google Analytics only capture actions related to a page visit. Direct data access are not captured, and bandwidth utilization is also not measured, however, Piwik and AWStats obtain those statistics.

Computing Backend

The computing backend for Health MAP can be handled by cloud services if the purchase and maintenance of a physical server are not possible. An 8 GB memory, minimum of two core processor, 1 TB Solid State Disks, and projected transfer volume of 3TB should be sufficient. As a strategy to ensure high availability, it is recommended that an alternate site is established that has a snapshot of the latest updates from the primary server. Details of building an alternative site are discussed in, Data Security and Risk Mitigation.

It is assumed that Cloud service providers, or server rooms for physical servers, are equipped with state of the art networking facilities and compliance with industry standards in cyber security. As a bare minimum, the server should be able to transmit data over a 1GB line (i.e., all 1G POE+ ports). Ideally, a network will be designed for speed, reliability, and capacity to handle the daily traffic needs of users, including the high speed and low latency bandwidth that users may require when transporting data. A 10G primary connection with 1G backup to commodity Internet and Internet2 is most ideal but not required.

Data Capture and Encoding

The procedures for data collection, recording, encoding and saving in a database management system are critical for data organization and discoverability. In the following, a unique record identifier is proposed that can also be used by software engineers to track and trace the origin of the file. A preliminary data schematic is presented, based on the draft data collection form and perceived supplemental forms related to clinical and laboratory examinations, when applicable. How the data will be exchanged with external systems, including NOAA's DIVER, USGS WHISpers, and NOAA's NSDB, is presented. Recommendations for the implementation of activities that support the open data policy of the government are outlined.

Persistent, Unique Record Identifier

Locating a record or a compilation of records is the primary function of any information system, and the definition of a unique record identifier is essential to many database designs. There are many different ways to create record identifiers. The most common technique is to provide a sequential number from a start point, very often starting with '1' and incremented by 1, for every record added to the system. As the information system grows, record identifiers of this nature are prefixed or suffixed with some additional label, e.g., nih-000123, 123-nih-green-21, to categorize the data in the collection. The syntax mutates and often, only the database administrator is aware of such changes. System documentation is often insufficient to describe the evolution of record identifiers. This non-standard creation of unique record identifiers makes it difficult to maintain the system in the long-term, especially when there is a change in personnel. Importantly, they do not allow for tracking of data in cases where a subsection of the raw data undergo additional processing or analyses, and in the process, generate new records. This scenario is typical in marine mammal health monitoring.

In computer science, the introduction of the Uniform Resource Name (URN) in 1997 [25] provides an option for identifying a resource, digital objects or records, that is location-independent. URNs represented in Backus-Naur Form (BNF), are case sensitive and can be a globally unique identifier that persists. Although URNs are not designed to be parsed to get information in a single string set of characters, they provide more information at a glance than a sequential set of numbers.

The following is a proposed syntax that can be followed for all records that are archived in Health MAP:

"urn:" <NID>":"<NSS>

The namespace identifier or NID is set to '**mmhmap**' (Marine Mammal Health Monitoring and Analysis Platform). In practice, namespaces should be registered with the Internet Assigned Numbers Authority (IANA)[26], a department of Internet Corporation for Assigned Names and Numbers (ICANN)[27] responsible for coordinating maintenance and procedures for namespaces. However, it is not required given that the intent is only for defining unique record identifiers within the information system. The same can be said about the use of the prefix, 'x-' to denote experimental namespace when an unregistered URN is used. The namespace-specific string or NSS can follow the following syntax:

<org>:<eId>:<aId>.<aIdn>:<lId>:<rId>.<rIdn>

where

<*org*> - an abbreviated label of the organization or individual reporting the event;

<*eId*> - local identifier used by the reporting <*org*>. If the <*org*> does not maintain a local identifier, label identifying the person recording the event, or a period the event started can be used and the format, yyy-mm-ddThh:mm:ssZ may be used (e.g., 2017-03-01T13:10:00Z) instead;

<*aId*> - this is the animal ID. If the animal identifier system is not maintained by <*org*>, a formatted string or sequential numbers, starting with '001' can be used;

<*aIdn*> - sequentially number from 1 to identify specific animals (default is '1') recorded during an event. The only time this number is incremented is if the <*aid*> returns to be recorded;

<*lId*> - this is similar to <*org*> but is in reference to the label for the laboratory where the tissue sample or other materials from <*aId*> were sent for analyses;

<*rId*> - this is the identifier given to laboratory results. If <*lId*> does not maintain the record identifier, a formatted string of sequential numbers may be used; and

<*rIdn*> - This is a sequential number starting from '1' that references a results document.

The following is an example of unique record identifier's application using the current data flow diagram (Figure 4.1), and how the proposed record identifier following the URN syntax, as suggested above, mutates as it goes through the process. It will be noted that the syntax will still provide enough information to trace the various samples back to the event and the animal involved in the event. It also allows records to be traced to all other related records. Figure 10 demonstrates the application of the syntax, and the following are assumed:

- The reporting organization is Fish and Wildlife Research Institute, and the label given is 'fwri'
- A tissue sample from both animals was sent to Key Biscayne Animal Laboratory (label: 'kbal') to analyze tissue samples, and a tissue sample was sent to RSMAS, University of Miami (label: 'um.rsmas') for further analyses.
- um.rsmas maintains a record of all laboratory analyses, and the next record label should read xx0015, and this is sequentially incremented. Also, it is assumed that um.rsmas only generates a page of the report.
- kbal also maintains a record identifier, and the next label should read 00131. Moreover, it is assumed that kbal creates multiple pages of the report (here assumed to be three pages for three specific laboratory analyses).
- The event involved two stranded dolphins and fwri maintains identifiers for recording such individuals. The last record on the fwri archive (not this one) was identified as 'pp-0012' and the next record, if we assume that fwri uses a sequential number for referencing records, should be labeled as 'pp-0013.'

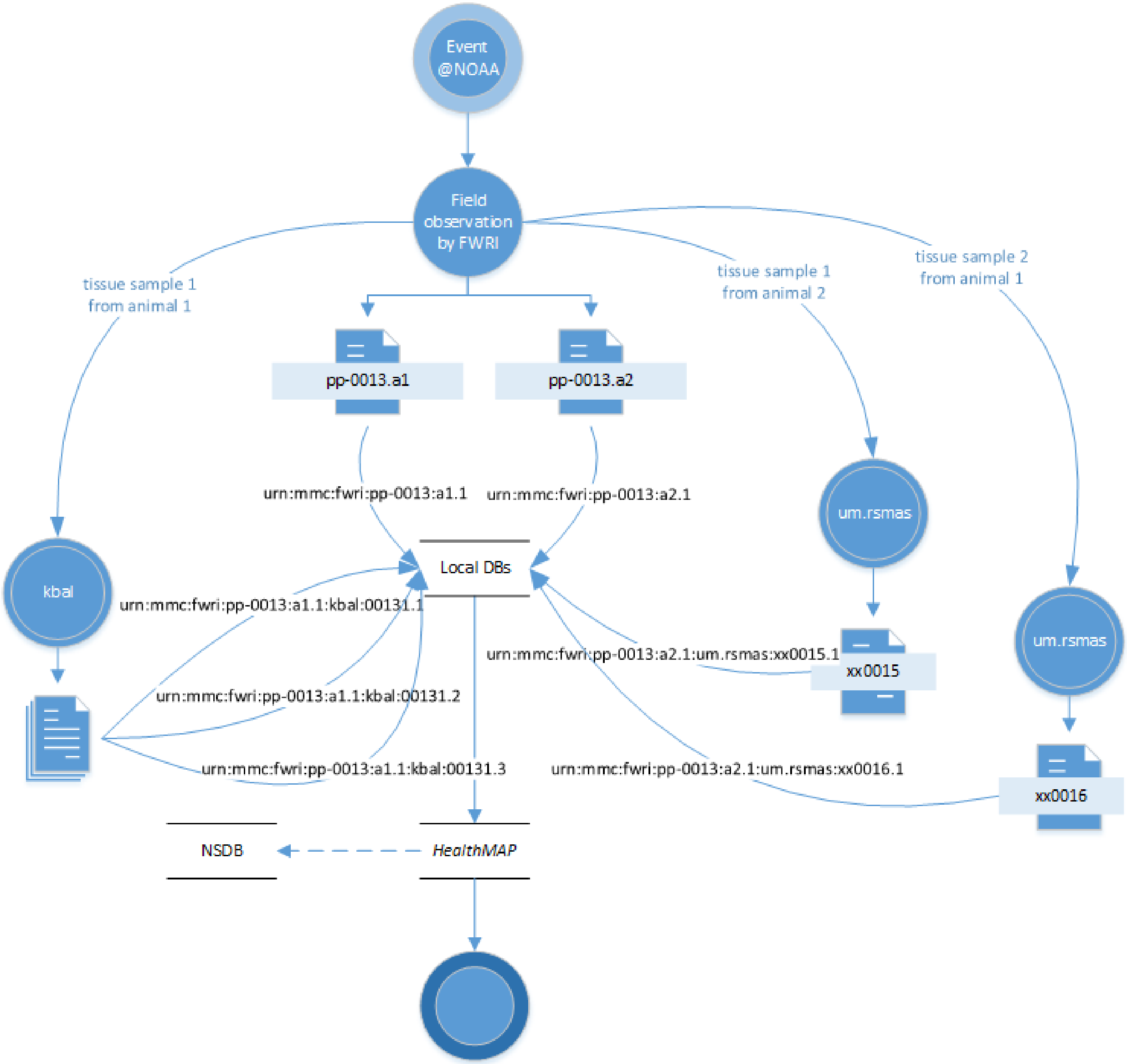

Event
@NOAA
Field
observation
by FWRI
tissue sample 1
from animal 1
tissue sample 1
from animal 2
tissue sample 2
from animal 1
pp-0013.a1
pp-0013.a2
urn:mmc:fwri:pp-0013:a1.1
urn:mmc:fwri:pp-0013:a2.1
um.rsmas
Local DBs
kbal
urn:mmc:fwri:pp-0013:a1.1:kbal:00131.1
um.rsmas
urn:mmc:fwri:pp-0013:a2.1:um.rsmas:xx0015.1
xx0015
urn:mmc:fwri:pp-0013:a1.1:kbal:00131.2
urn:mmc:fwri:pp-0013:a1.1:kbal:00131.3
urn:mmc:fwri:pp-0013:a2.1:um.rsmas:xx0016.1
xx0016
NSDB
HealthMAP

Figure 10. Schematic representation of the data flow and how Health MAP record identifiers are mutated and formatted.

The recommended approach requires that organizations maintain a record of events and animal identifiers for each event. Sequential numbering is sufficient to apply the recommended syntax. If a simple record number is used, sequential or otherwise, that combines the event and animal being observed, the additional identifiers should be generated (see the recommended module on a centralized generation of identifiers) to identify the event and animal.

Data Schema Development

Appendix B is a preliminary data schema for the basic animal information based on the proposed new version of the data collection form as drafted by Health MAP working group. These schemas can be used to define tables in databases and XML files for data exchange. It is important to note here that as the form is developed, this data schema should be updated or extended to support other needs of the program. The preliminary data schemas (Appendices B to G; Table 8) are hosted by GCOOS (Namespace: http://data.gcoos.org/mmhmap/xml/1.0). Figure 11 is a dependency relationship diagram of the table components that supports Health MAP, describing how other elements of the system supports the basic animal information database.

Table 8 . URLs of preliminary data schemas for Health MAP.

Data	Schema (http://data.gcoos.org/ mmhmap/xml/1.0/)	Documentation (http://data.gcoos.org/mmhmap/)
Basic Animal Information	basicAnimalInfo.xsd	MMHMAP_basicAnimalInfo.pdf
Group Events	groupEvent.xsd	MMHMAP_groupEvent.pdf
Laboratory examinations	animalLab.xsd	MMHMAP_animalLab.pdf
Necropsy details	animalNecropsy.xsd	MMHMAP_animalNecropsy.pdf
Released details	animalRelease.xsd	MMHMAP_animalRelease.pdf
Euthanized/Died details	animalDied.xsd	MMHMAP_animalNecropsy.pdf

There are existing online databases that can be used to supplement Health MAP to provide species taxonomic nomenclature and details. A good example of this is the extensive database of marine mammals maintained by the World Registry of Marine Species (WoRMS) (http://www.marinespecies.org /cetacea/). It is recommended that a direct link to WoRMS' information system be established to obtain proper scientific name labeling and other taxonomic information about the species (Genus, Family, Order, and Class). The taxonomic labeling is crucial and needs to be standardized in Health MAP to ensure that group summaries of species can be performed and will allow for seamless integration with external information systems that are in compliance with WoRMS' listing.

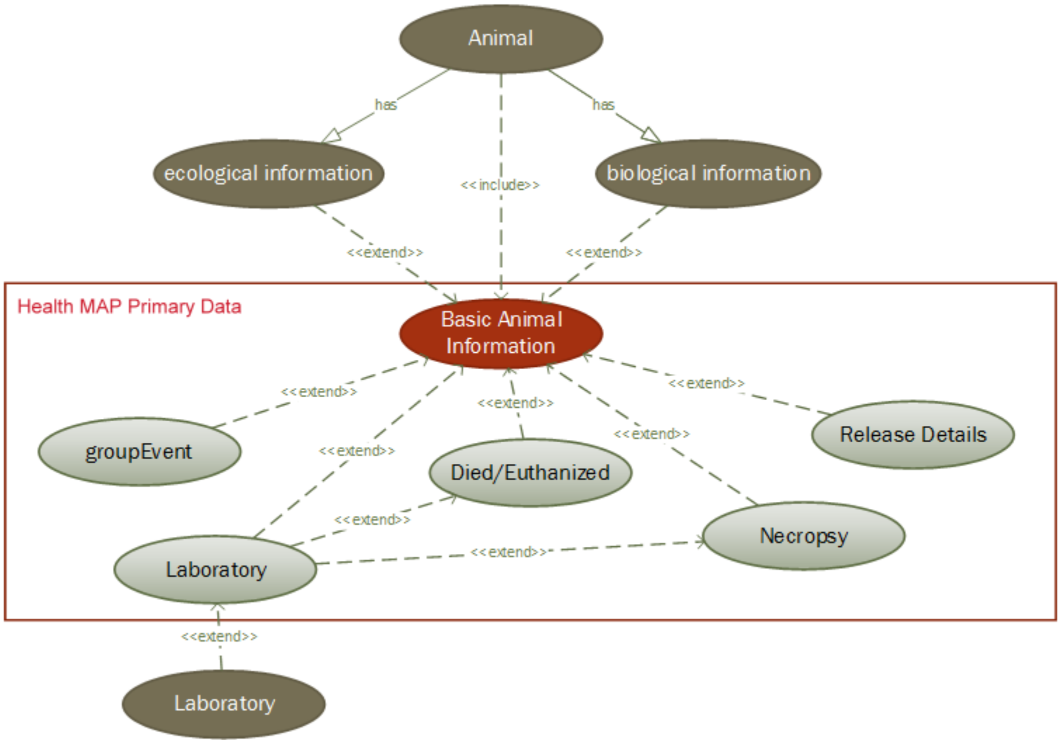


Figure 11. Dependency relationships among the various table components of Health MAP. Not all information resides in Health MAP's Basic Animal Information table. The tissue analyses and other laboratory results are treated as a separate data element.

Notes on Synonyms

Other online databases, such those maintained by the Smithsonian National Museum of Natural History (http://vertebrates.si.edu/mammals/mammals_databases.html) can provide additional information about an animal. However, they should be avoided as a source to validate taxonomic names given that the system does not maintain synonyms. As taxonomic nomenclature undergoes regular review, the taxonomic names can change and link back to proper labels can only be made via the registered synonyms.

It is a not a requirement for Health MAP to describe the ecological details of the species. However, a link to a system that can provide other biological, morphological, and ecological details (e.g., environment, range and distribution, maturity indicators, life cycle and mating behavior, feeding behavior, and diet composition) will make the information system robust and informative beyond just reporting events.

It is important that the laboratories engaged in Health MAP are registered and fully qualified laboratories to perform autopsy and tissue analyses of marine mammals. The laboratories should submit reports in compliance with requirements, such as the use of the URN for Health MAP records and other standards that may emanate later. The data to be collected from laboratories or the procedures for laboratory examination is still in discussion. However, once this is decided, most of the laboratory results are best submitted in a free form, i.e., remarks field. Searching via free-form fields is not a problem in modern database management systems but can tax the system if there are too many fields to search. The laboratory records expected for Health MAP are considered small and hence, should not pose a problem. Trying to establish a structure for laboratory results is not advisable, as it tends to restrict procedures for a full forensic or diagnosis of the specimen or carcass submitted following an event.

Data Exchange and Interoperability

Figure 8 presents an architecture that allows other information systems to have machine-to-machine (M2M) access through a RESTFul service endpoint, a URI that accepts web requests. Communications between internal components, as well as, external systems such as NSDB, WoRMS and NOAA DIVER will be done largely via XML messaging using a data schema (Figure 12).

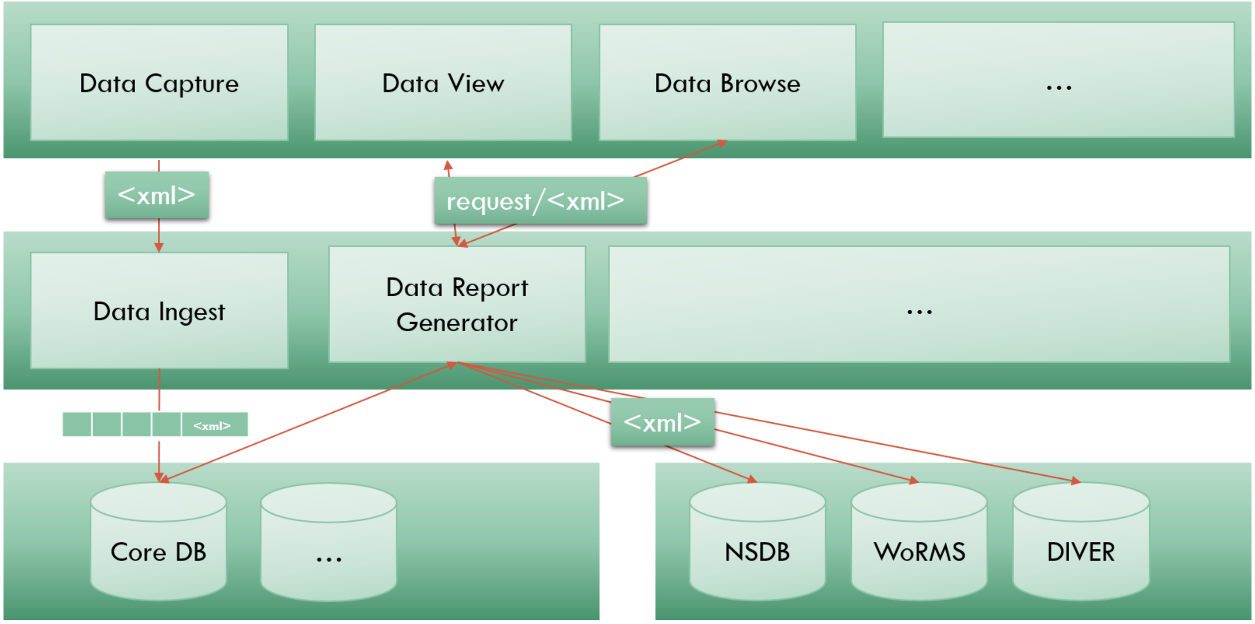


Figure 12. Data communication and transport in Health MAP.

The application of Darwin Core (DwC) is recommended to effect a seamless integration with other information systems. A vast majority of the DwC terms do not have an equivalent to the fields or variables described in the data schemas (Appendices A to G). However, this does not mean that it should be abandoned. The absence of terms used in marine mammal health monitoring activities in DwC is an opportunity for Health MAP to contribute to the collection if long-term interoperability is to be addressed. The Darwin Core Resource Description Framework (RDF) Guide [28] should be consulted to allow Health MAP terms to be shared using RDF (see Standards and Vocabulary).

Encoding and Updating List Boxes

In the previous chapter, it was proposed that the Data Services package should provide services to update the drop-down lists. The lists that comes with the list boxes should be considered and treated as data inputs. However, the lists are consensus from Health MAP working groups rather than just an arbitrary input from selected individuals. In addition, it is important that changes to the list are backward compatible or have a one-to-one correspondence with the previous list.

Figure 13 is a proposed workflow that can be followed for changes to the drop-down list or any other parts of the database (see SDA Change Management).

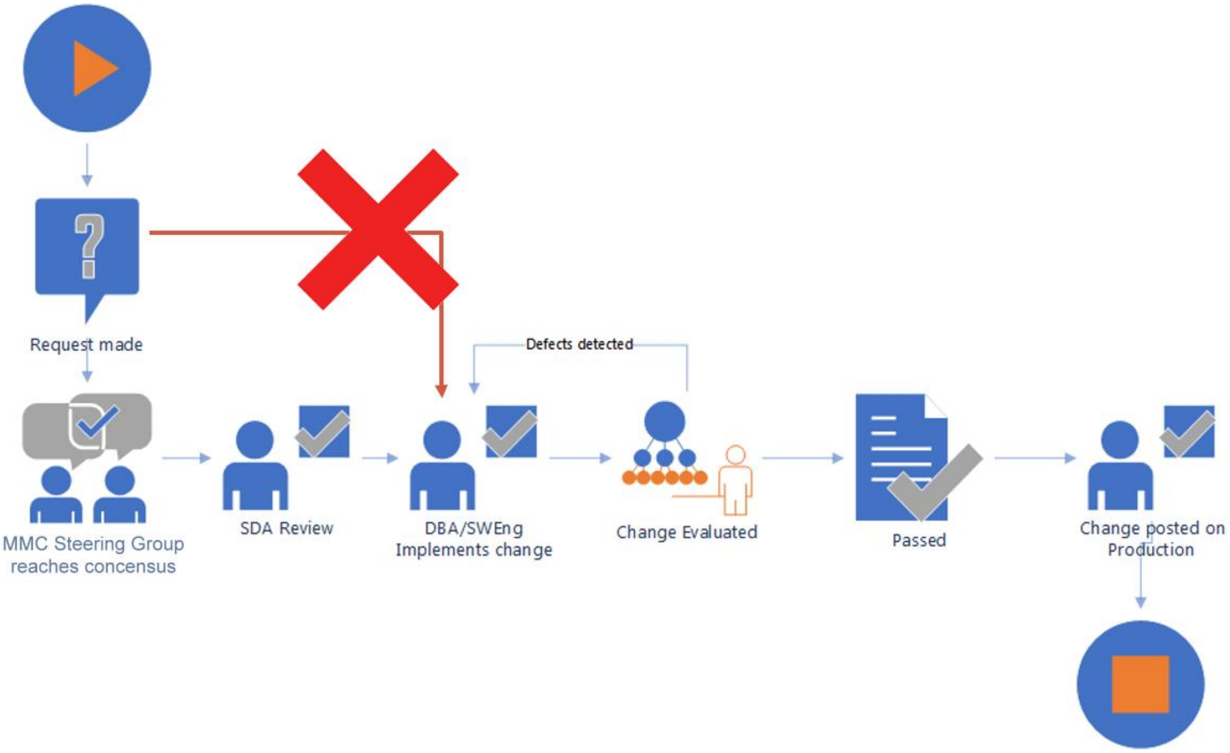

Request made
Defects detected
MMC Steering Group reaches concensus
SDA Review
DBA/SWEng Implements change
Change Evaluated
Passed
Change posted on Production

Figure 13. Proposed workflow for changes to the selection list or any part of the system that influences data to archive.

The Database Administrator (DBA) and the Software Engineer (SWEng) are integral in the change process. A change in functional requirements can lead to a chain of reactions that may require a change in the database structure and can affect the forms used in collecting data, and reports generated by Health MAP. In addition, the information system will need to undergo regression testing to ensure that the other parts of Health MAP are not affected by the change.

NOTE: It is NOT recommended to bypass a step in the workflow (Fig. 13) to introduce a change in the information system. Submitting a change request directly to the DBA or Software Engineers should be avoided. More on change management is discussed in SDA Change Management section of this document.

Data Security and Risk Mitigation

The integrity of the system is best maintained with consistent and well-established data access protocols and by mitigating the risk of data loss with recovery strategies. Health MAP has established resource security protocols, and the following are recommendations that can be employed to avoid data loss or corruption, and minimize downtime of Health MAP services.

User Account Management

There exists a Health MAP Data Policy (Appendix H), and this should always be the basis for establishing user access to the data. In the most recent version of the document, four levels of data access are proposed.

General Public (Level 1)

The general public can access summary data of an event or incident that includes Basic Animal Information (BAI) identifying the species, date, location, data provider and a summary of the health condition/category. Available to anyone browsing the Health MAP portal without registering or logging into the system. Registered users can download and get more details about the record. Anyone may register to become a user of Health MAP data, by providing some basic information about themselves, their interest in Health MAP data, and registering agreement with the data use policy. All user information provided to Health MAP during registration will be kept for internal reference with Health MAP only. Data users are expected to acknowledge and attribute the data source to Health MAP as well as specific data providers, where requested, for specific datasets. The recommended data citation, when applicable, accompanies the downloaded data.

Notes on Citation

All data, processed or raw, plotted, mapped or downloaded as a table, should always be accompanied with the recommended data citation format. Health MAP Steering Group should establish the data citation syntax.

It is important for users to either acknowledge a disclaimer prior to data download or included in the data downloaded. A possible disclaimer may read:

"The data provided in Health MAP is verified and validated to the best of our ability. Health MAP and its affiliated agencies, partners and collaborators bear no responsibility for negative outcomes associated with the use of any Health MAP data."

Full Read-Only Access (Level 2)

This level of data access is for registered users to have read only access to BAI and detailed health data for all records that have been 'shared with to all registered users'. This includes records/data that have voluntarily been made available as well as those that are >1-year-old and available under PARR or other requirements. In an emergency, responders would be granted Level 2 access to ALL records in Health MAP (whether shared with all registered users or not). In rare cases, authorities, such as emergency responders, are provided with Level 2 data access for an unrestricted read access to the records.

It is recommended here that Health MAP has pre-created user accounts that can be distributed in cases of emergency to responders. These accounts can be reset (i.e., change passwords) manually or via an automated script. In many instances, the account's password is automatically altered after a single use. This rule can be relaxed to extend the life of an account. Other options include a 30-day access after the first use. The latter is the preferred option as it limits the number of communications needed to give emergency responders access to Health MAP.

A variant of this data access level, although not recommended and should be avoided, can be provided to certain individuals where full read-only access to health records will be allowed, and the password does not expire.

Researchers/Collaborators (Level 3)

Researchers and collaborators should be given a Level 3 access. This level has the same read access as Level 2 and read access to datasets they are collaborators on that may not yet be shared with all registered users, and full read/write access to their own data sets. Registered data providers and data diplomats from where the data emanated will automatically be given a Level 3 access to the repository. When applicable, read-only data will be anonymized.

Note on Data Anonymization

Anonymizing data should be avoided whenever possible. Anonymizing data requires guidelines to be established and regularly reviewed, and importantly, requires resources to implement the anonymization that are often not available.

Administrative (Level 4)

Full read/write access to all records and repositories (users and laboratories) should be given to selected few. More than one person should be provided a Level 4 access to the information system for maintenance (updates, patches, data management, security, account management) to avoid accidental loss of access.

WARNING! In cases wherein an account with Level 4 access has to be deleted, a protocol should be established to reset ALL Level 4 user accounts to avoid accidental security breaches.

Backup and Restore Network

Network communications and hardware can break or databases corrupted. It is recommended for Health MAP to follow the procedures described below to ensure a complete system recovery and limit service downtime. Figure 14 is a proposed network architecture that will ensure data and the information system, in general, can be recovered in case of loss (data and communication). The 'production' level server is the point of communication between Health MAP and the users. The 'development' server is the server that software engineers use to develop, update, and test modules before they are deployed to the 'production' server. There may be other servers in the local network, such as dedicated HTTP/FTP server used for other purposes, which are given access to Health MAP via the 'production' server. These servers are considered nonessential and will not be covered in this document.

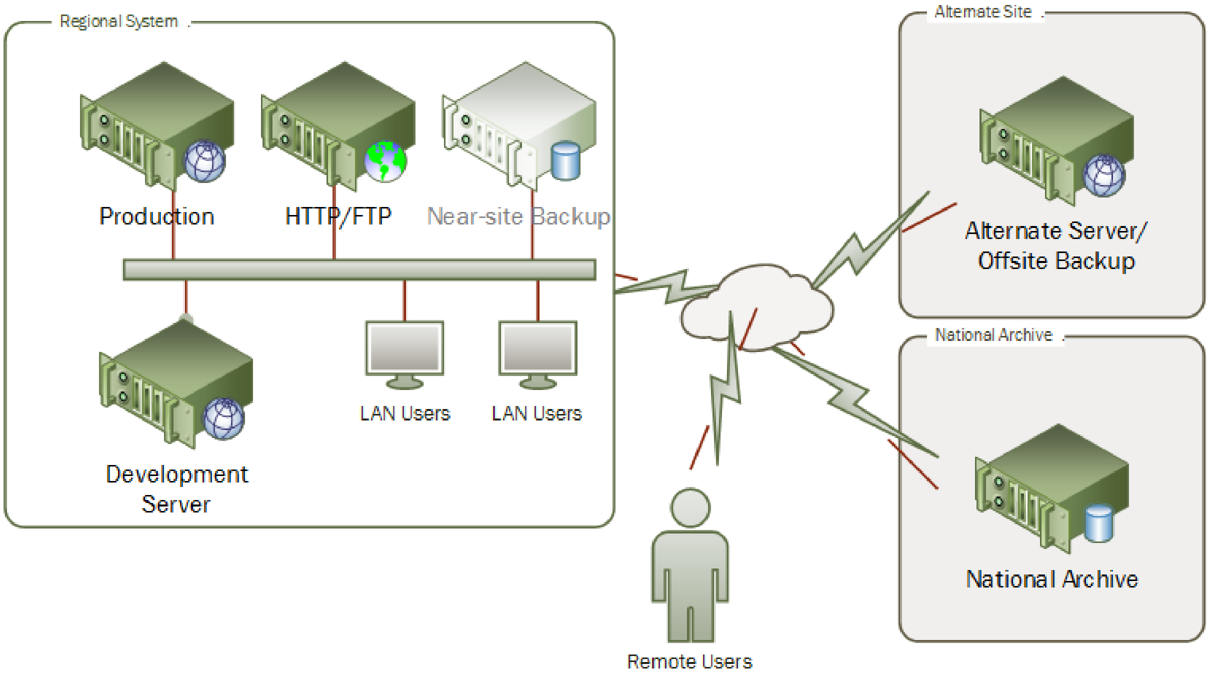


Figure 14. Proposed network diagram to limit the risk of data and communication loss.

On-Site Backup

An On-Site Backup system is simply the creation of a folder in the same server/computer that system managers can use for quick recovery. This server can contain a daily, weekly and monthly mirror of all codes and databases (using the rsync command for Linux-based systems is recommended).

Near-Site Backup

If a quick recovery is required, but the On-Site Backup is not available, system administrators can use "Near-Site Backup" resources. Very often, this is just another server in the same network that can share a space. Given the projected size of Health MAP (i.e., it is not expected to grow over 3GB in the next five years), a daily, weekly and monthly synching of the resources is recommended, as is recommended for On-Site Backup.

Off-Site Backup and Alternate Server

Maintaining an Off-Site Backup is highly recommended to ensure high availability of services. Disruption of network communication is widespread, most especially during a severe storm or other natural calamities. Disruption is also experienced when the security of the system is breached. If a mirror site is available from outside the network and geographical location (i.e., off-site), the information system can be recovered, and services re-established with ease by re-syncing the production server from the alternate site.

DNS Rollover

In cases wherein a prolonged disruption is expected on the production server, the Off-Site Backup, should be configured as the alternate server. The application of a Dynamic DNS (DDNS) is recommended wherein the domain is automatically rolled-over to the alternative site (Figure 15). Manual rollover can also be performed in cases where the DNS provider does not have that capability.

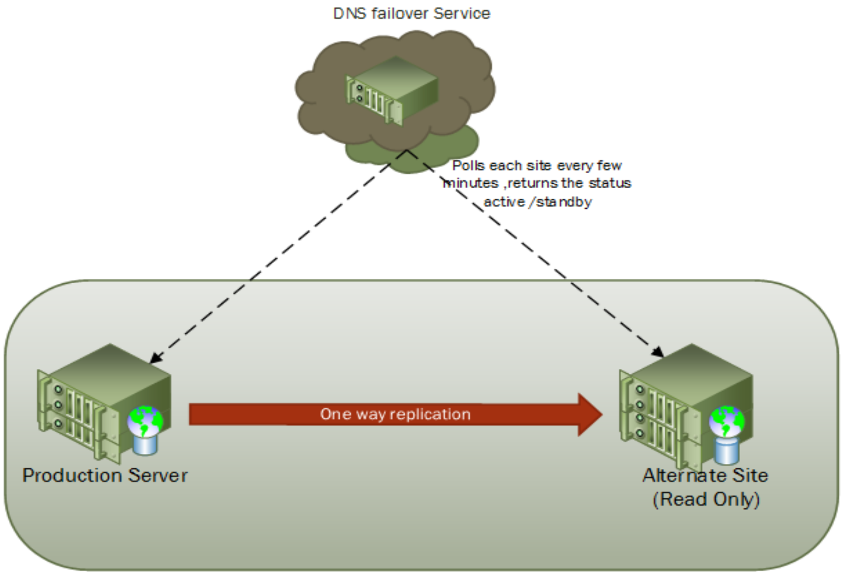

DNS failover Service
Polls each site every few minutes ,returns the status active /standby
One way replication
Production Server
Alternate Site (Read Only)

Figure 15. Rollover and synching of the alternate server.

The domain change will be transparent to the users, but it is not recommended to allow data providers to update or add records until all services are returned to the 'production' server. This restriction will prevent data loss as the Health MAP 'production' server returns to service (note the proposed one-way replication).

Disaster Response

In many organizations, a protocol is in placed to respond to disasters (e.g., loss of data, database corruption, loss of network communication, security breach). If the Health MAP host organization does not have a workflow to respond to disasters, Figure 16 is a commonly used workflow that can be instituted as soon as possible. This workflow requires Health MAP to establish a Disaster-Recovery (DR) Team that will meet (remotely or otherwise) to discuss who, when, how stakeholders will be notified, how to assess the damage and what strategy is best to recover.

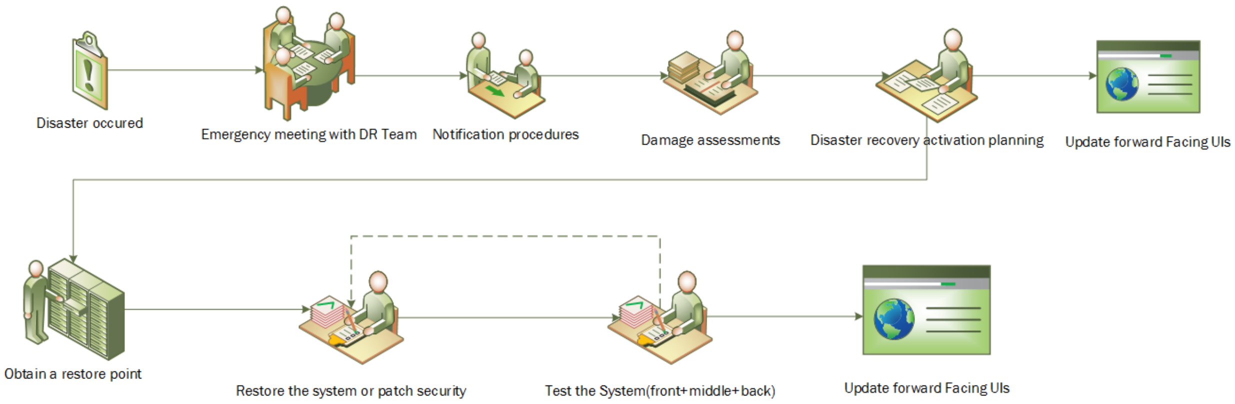

Disaster occured
Emergency meeting with DR Team
Notification procedures
Damage assessments
Disaster recovery activation planning
Update forward Facing UIs
Obtain a restore point
Restore the system or patch security
Test the System(front+middle+back)
Update forward Facing UIs

Figure 16. Proposed workflow in responding to a disaster.

There is no one-size-fits-all workflow. However, it is essential to keep the public notified of any disruption of services or when services return to normal. All incidences of service disruptions should be noted and recorded for future references. When applicable, the journal should also include a copy of all server-based system logs for technical references for future forensics when needed.

Vocabularies and Standards

Interagency or inter-organizational efforts or activities have become very common, and with them, comes the sharing of data. Interoperability of information systems, locally, regionally or internationally, is not limited to the employment of the same data structure or data types. The use of community standards and application of controlled vocabulary is paramount in making the system and the data within it understandable and shareable across systems, thereby reducing operational cost and leveraging existing resources and investments.

Controlled Vocabulary

A Health MAP working group has started the compilation of terms used in monitoring marine mammal health. It is not expected for the group to complete the definition of all terms, and the list(s) should be taken as a living document. Getting consensus on a definition for a term can be challenging. Maintaining two lists, one that has been agreed unanimously and another that will require more discussion is recommended. The definition and labeling terms that the group have agreed is the first of many steps leading to a controlled set of vocabularies. The list(s) should not be saved in a table or relegated to a publication. In the context of Health MAP, these terms should be aligned with community standards and registered to create a persistent reference.

Registering and Darwin Core

Darwin Core (DwC) is a body of standard terms. It is a glossary of terms and their definitions to facilitate data and information exchange. Initial mapping of the terms used by Health MAP reveals that most of these terms are not in DwC. To submit these terms to DwC, the following steps are recommended:

Step 1. Export the Health MAP definitions, i.e., those that have been agreed upon, to an RDF (see Appendix I for example). Note that it is hard to satisfy the definition of all the terms used in Health MAP. For this step, focus on the terms that have already been agreed.

Step 2. Import these term into a registry such as the Marine Metadata Interoperability Ontology Registry and Repository (mmisw.org), Federation of Earth Science Information Partners (ESIP) Community Ontology Repository (cor.esipfed.org), or NSF Cross-Domain Observational Metadata for Environmental Sensing (X-DOMES) Ontology Registry and Repository (https://xdomes.org/ont). A test deployment of the RDF files generated (Appendix I) is posted at https://xdomes.org/ont/mmhmap/lab_procedures.

Step 3. Map the terms to DwC and submit the Health MAP terms for inclusion and adoption by DwC.

Step 4. When DwC has adopted the terms, change all reference to point to DwC but maintain the registry of the terms.

Health MAP terms should be treated as a living document and will require a continuing review. An annual workshop to review the terms is recommended to ensure that they remain valid and in accordance with community practices.

Data Schema and OGC O&M

Having a controlled set of terms is only part of the effort necessary to make the system interoperable with others. The way the data are shared across systems is as important as the definition of the terms (above). Very often, an information system provides endpoints from where the databases can be queried and results downloaded. The downloaded or shared data, if not following a community standard, will have to be read by developing a script or codes to read and parse the data correctly specific for an endpoint. However, if it follows a community standard with recognizable data schema, the process of reading the data will be straightforward and easy. The guess-works needed to decipher what the returns are or what units of measure were used are eliminated.

There are a number of standards development groups like the Institute of Electrical and Electronic Engineering (IEEE), International Organization for Standardization (ISO), American National Standards Institute (ANSI), and Open Geospatial Consortium (OGC). OGC provides some standards that can be extended to incorporate the parameters recorded in Health MAP. The preliminary proposed Health MAP data schema, can be used immediately for data exchange.

However, the wider adoption of the data schema will be inhibited by the lack of fair process and community involvement by which it was formed. OGC has a proven process from conceptual formulation to community adoption. Engaging OGC from the outset will facilitate the community adoption of the data model. If adopted, it will greatly facilitate data exchange between information systems; and hence, interoperability.

The process from the beginning to community evaluation, to OGC adoption, can take three months to a year or longer depending on the topic and community involvement. Care should be taken in incorporating the schedules, and dependencies to the OGC adopted data schema. The advantages and benefits of having and following a community-wide standard outweigh the time needed to get community approval.

Organization and Change Management

To sustain Health MAP, an organization structure needs to be established to support its continuing development, application, and deployment. Figure 17 is a proposed organizational chart to support Health MAP.

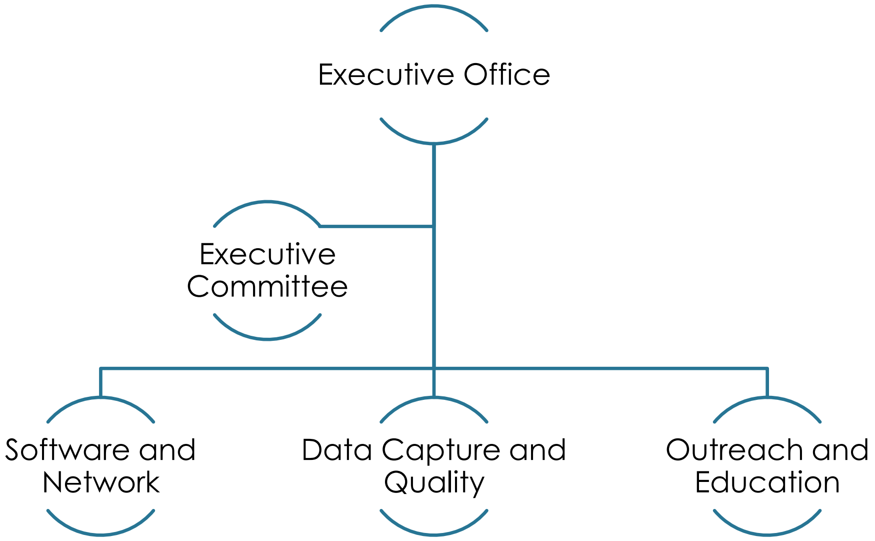


Figure 17. Proposed organizational structure to support Health MAP.

Executive Office

The Executive Office (EO) coordination of all activities related to Health MAP, including software package continuing updates and development, data collection and reporting, data access protocols and distribution, and product disseminated. The EO can be composed of a National Coordinator (NC), preferably reporting to and coordinating actions with the Executive Committee. Part-time secretarial support is recommended to assist the NC in the coordination of activities with the Health MAP program and with stakeholders.

Executive Committee

The Executive Committee (EC) is required to ensure that directions and strategies employed to mature Health MAP remain in compliance with organizational protocols and practices. The EC should include, among others, a representative from the private sector to help guide management for the public adoption of the products. The EC should not be limited to individuals working on marine mammals but should also include ecologists, physical scientists, and natural resource managers.

Software and Network

This unit supervised directly by the NC will be tasked to manage the infrastructure that will include, software, hardware and network maintenance and upgrades. Given the project size of Health MAP, i.e., estimated to grow no larger than 3GB in the next five years, only one person may be required to manage the system, unless a massive software update/upgrade is required. In which case, subcontracting the additional task may be required.

Data Capture and Quality

This unit supervised directly by the NC will evaluate the data as they are continually captured. Also, this unit will be tasked with assessing and evaluating data collection procedures, and what additional data are required to enrich the collection.

Outreach and Education

This unit, supervised directly by the NC, will be tasked with designing programs for continuing education, i.e., training user, certifying trainers and as a help desk for all users. Moreover, this unit will evaluate training materials and manuals, to remain current with changes to the data collection procedures, software, and user interfaces.

Change Management

It is imperative that a ‘change management’ process is followed when a change is required in Health MAP. Figure 18 is a typical workflow to process a change request. If a change is needed, regardless of how small the change is, it should go through this process. The process starts with a request and the NC will organize a group to discuss and accept or reject the request. When a consensus is reached, the change request will be evaluated with the latest version of the SDA to ensure that the change will not conflict with other components of the system.

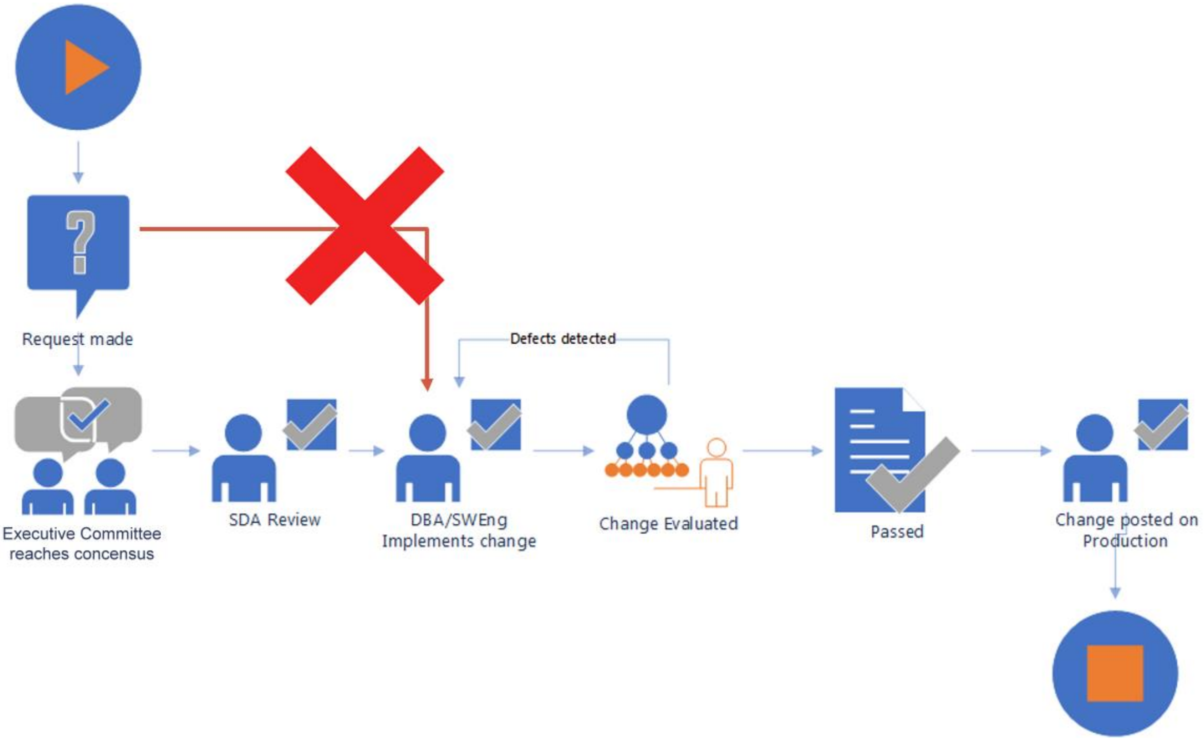

Request made
Defects detected
Executive Committee reaches concensus
SDA Review
DBA/SWEng Implements change
Change Evaluated
Passed
Change posted on Production

Figure 18. Recommended workflow to change something in the system.

Once a change is approved, a job is created for the Software Engineers or Database Administrator to implement. The initial product will undergo testing and evaluation. The testing and appraisal procedures will be dependent on the change made. The application of software tools to do regression testing is highly recommended. If the change does not introduce problems or error during the testing phase, the codes will be uploaded to the production server to update the system. The SDA will also be updated only after the successful test and evaluation, i.e., before uploading the updates to the production server.

Directly approaching the Software Developers or Software Engineers to change something in the system (add, modify, delete), or bypass the SDA review process, should be avoided. Failure to follow protocols to effect change, e.g., review the change request with the architectural document, often leads to more problems or even to system failures or instabilities.

Tasks and Module Implementation

The following is a Timeline and Gantt chart of the recommended activities (tasks) for the development and deployment of the proposed Health MAP. There is two subsections in this document: (1) A-to-Z project, and (2) modularized project tasks.

A-to-Z Project

The "A-to-Z Project" is the development and deployment of the proposed design and architecture as a 3-year project.

1. Project Scoping, Visioning, Team Building

This task includes the preliminary activities to establish a shared understanding of the goals and visions of the project. A 2-day Project Scoping, Visioning, and Team Building (PSVT) workshop are recommended to form an efficient team. Team members should know their terms of reference, responsibilities in the project, risk(s) related to their performances, and knowing who the other members of the team are, is paramount to the success of the project.

This task will have four sub-tasks: (i) pre-PSVT to prepare for the workshop, (ii) PSVT workshop, (iii) finalization of the project scope based on inputs from the PSVT workshop, and (iv) securing core resources, such as workspaces, and workstations.

2. System Backend and Database

This task is considered one of the first major tasks of the project. It includes six sub-tasks:

Server initialization and installation

This subtask is to secure the servers as recommended and installation of required software packages needed to execute the proposed information system. As recommended, Health MAP should be executed from a Linux-based enterprise operating system such as CentOS. The primary packages may include, Python 2.7/3, PostgreSQL, PHP 7, OpenSSL, Apache HTTPD, Tomcat 7, fail2ban, Java 1.8, and OpenSSH6.6. The latest stable releases of all these software packages should always be considered as they can include security patches.

The standard TCP ports expected to be enabled on the network firewall are ports 22 for SSH, 80 for HTTP, 8080 for Tomcat, 25 for SMTP, 443 for SSL, and 5432 for PostgreSQL. It is not just the server that will house Health MAP that will need to be initialized, but system development and management environments as well:

(i) initialization of a GitHub or BitBucket repository to centralize the codes in OpenSource environments,

(ii) issue monitoring system using Trello technology or Atlassian Jira,

(iii) team collaboration workspace such as Atlassian Confluence, and

(iv) integration and release management system such as Atlassian Bamboo.

Data schema review

This subtask is to examine the preliminary data schema as proposed here. The data type and data length or precision (integer, floating) are attributes that may need additional scrutiny. Note the proposed use of the persistent record identifier as the primary key whenever possible.

WARNING! It is very easy to get stuck in discussions on the data type and size. It is recommended to use varchar data type for strings, and floating data type for numbers when uncertain. Data structures can be optimized in later stages of development, or as the need arises due to late changes to the system requirements. The information system is not expected to hold an enormous amount of data; hence, optimizing the table designs for speed to access data, is not relevant. The draft data schema (http://data.gcoos.org/mmhmap/xml/1.0/) include data attributes that should be considered as a starting point for discussions.

Schema to OGC O&M

The proposed data schema, while it can be used as is for Health MAP, it is highly recommended to establish standards or community acceptance that a wider group of users (e.g., scientists, researchers, data users from other discipline or countries) can employ (see section on Vocabularies and Standards). The Open Geospatial Consortium (OGC) is one of those groups that maintains standards such as the Observation and Measurements (O&M). The O&M model is a natural place for parameters measured and recorded by Health MAP.

This subtask will include the formation of a team lead by OGC to extend the O&M standard, a workshop to get the team to discuss and exchange thoughts and draft the new standard, publication of a test server and XML-based storage of data to demonstrate the extended O&M standard. This test server will remain open for community testing and evaluation. This task will culminate in the adoption of the extended O&M by OGC. Provisions should be made to ensure that this new data standard will be revisited by representatives of the community to ensure that the terms remain current.

Vocabulary

This task will focus on continuing efforts to establish a common vocabulary for Health MAP. Existing products from GulfMAP development will be used as a starting point. While it is not expected to reach unanimous agreement on all the terms, those that the community agrees will be mapped to Darwin Core (DwC) terms. If the term is not included in DwC, these terms will be submitted to DwC for consideration. The end product of this subtask is a Resource Description Framework (RDF) compliant files to facilitate the registration of the new terms in a registry such as the Marine Metadata Interoperability Ontology Registry and Repository (https://mmisw.org/ont) that can be used as a persistent reference for the term.

Backup and Recovery

In reference to the section on Data Security and Risk Mitigation (above), this task is to establish and document the on-site, near-site and off-site backup systems. It will also include installation of an automated system to roll over the domain, i.e., configuring a Dynamic Domain Name Server (DDNS), to an alternate, off-site server. The rollover function is to ensure high-availability of the services in cases where the services and communications to and from the primary server are disrupted; the alternate server is activated.

A comprehensive guide will also be written, detailing a tested restore procedure from all sources. Steps will be taken to ensure that the backup and restore procedures are the most efficient and remain effective to restore Health MAP when needed.

System Integration Test and Review

This subtask is to examine/evaluate all services developed from all the other subtasks. The task is also to build or script an automated system to continually test, monitor and alert individuals, as described and recommended in the System Alerts and Messaging package of Health MAP.

3. Data Ingest

This task will commence with a thorough review of the functional requirements as stipulated in the SDA. The review of technical requirements should include representatives from stakeholders, most especially, data providers. During the review process, any adjustments or changes needed in the original list of functional requirements will be discussed and approved by the technical lead to ensure that the various components will still work as designed and scheduled. The application of the 'Agile/Scrum' development approach, the use of issue tracking systems, and open source facility that should have been done in Task 2 above are highly recommended. Modular, as well as system integration testing, is required. What was tested, how it was tested, and the test results should be documented and submitted as a report that completes this task.

All data should be stored in PostgreSQL (or equivalent) combinatorial databases with XML fields using the data schema for Health MAP or No SQL-based system like MongoDB*,* which allows for concurrent use via the Internet. This subtask should culminate with documentation of the package containing the technical details, comprehensive user guide, and parameters for proper regression testing.

4. Data Report Generator

Like Task 3, this task will commence with a thorough review of the functional requirements as stipulated in the SDA. The review process should include representatives from stakeholders to ensure that the exposed logistics behind the web services can easily be understood. During the review process, any adjustments or changes needed will be discussed and approved by the technical lead to ensure that the various components will still work as designed and scheduled. In reference to the functional requirements of the Data Report Generator, this module should develop modules with the capability of reading data from Health MAP and other external sources (see sections Health MAP Proposed Architecture, page 8*,* and Data Capture and Encoding, page 24), and importantly, convert the data to follow the proposed data schema. This subtask should culminate with documentation of the package containing the technical details, user guide, and parameters for proper regression testing and use by other modules in the system.

5. Authenticator

Like Task 3 and 4, this task will commence with a thorough review of the functional requirements as stipulated in the SDA. During the review process, any adjustments or changes needed in the list of functional requirements will be discussed and approve by the technical lead to ensure that the various components will still work as designed. In reference to sections on Data Security and Risk Mitigation (page 33), the module to provide persistent session will be developed and deployed by this task. User authentication provided by an OpenID technology provider, such as Google, LinkedIn, and Facebook, should be considered.

This subtask should culminate with documentation of the package containing the technical details, user guide, and parameters for proper regression testing and application.

6. Data Browse

Like Task 3, 4 and 5 above, this task will commence with a thorough review of the functional requirements as stipulated in the SDA. During the review process, any adjustments or changes needed in the list of functional requirements will be discussed and approve by the technical lead to ensure that the various components will still work as designed. This module will work with the Data Report Generator where it will send the data request via the RESTful web services of the Data Report Generator. The XML-based result will be parsed and presented in a configurable table form and should be exportable to a CSV, TSV or PDF.

The configurations for the user-defined tables can be named and saved in a database that can be shared to a common repository for future references or for sharing with others. Note that since this is a package in the Application Layer of the architecture, the modules that will be developed to support data browsing should be 'friendly' to mobile devices.

This subtask should culminate with documentation of the package containing the technical details, user guide and parameters for proper regression testing.

7. Data View

Task 7 is similar to Task 6 (above), but results will be presented in graphical form rather than tabular. Like all software development tasks above, this task will commence with a thorough review of the functional requirements as stipulated in the SDA. During the review process, any adjustments or changes needed in the list of functional requirements will be discussed and approve by the technical lead to ensure that the various components will still work as designed and scheduled.

The graphical outputs will include a user-interactive and user-configurable online mapping system, and when appropriate, user-interactive graphical representation of the data. This task includes graphical representation using polar, bar and line graphs to show the holdings. The interactive plots and maps should also allow the user to slide through a time scale interactively and results be presented automatically.

Like Task 6 above, this module will work with the Data Report Generator where it will send the data request via the RESTful web services of the Data Report Generator. The configurations for the user-defined request can be labeled and saved in a database that can be shared to a common repository for future references or for sharing with others. Note that since this is a package in the Application Layer of the architecture like the Data Browse, the system should be mobile-device-friendly. Given the limited resources (storage, memory) of mobile devices, not all functions may be available when the Data View module is accessed via mobile devices. Functions or services that will be disabled on mobile devices, should be listed and noted on the functional review document created at the start of development.

This subtask should culminate with documentation of the package containing the technical details, user guide and parameters for proper regression testing.

8. System Alerts and Messaging

Like previous tasks, this task will commence with a thorough review of the functional requirements as stipulated in the SDA. Involvement of stakeholders at a high level is essential for the success of this task. During the review process, any adjustments or changes needed in the list of functional requirements will be discussed and approve by the technical lead to ensure that the various components will still work as designed and scheduled.

This task will develop modules that can be used by all packages in the system. It should have the capabilities to capture all messages, either it is a status report or system defects or failures (i.e., system errors), and to forward them to appropriate endpoints (e.g., email to administrator, data providers). System users (persons and packages) should have access to a user-configuration module to define the kind of message(s) they want to receive. A user with administrative functions may use the system messaging services to alert users about updates or system notices as the need arises. Note that since this is a package in the Application Layer of the architecture, the system should be developed to be mobile-device-friendly.

This subtask should culminate with documentation of the package containing the technical details, user guide and parameters for proper regression testing.

9. Data Services

Like the previous tasks, this task will commence with a thorough review of the functional requirements as stipulated in the SDA. Involvement of stakeholders is essential for community adoption of Health MAP. During the review process, any adjustments or changes needed in the list of functional requirements will be discussed and approve by the technical lead to ensure that the various components will still work as designed.

The Data Services should include, but are not limited to:

- Persistent record identifier generator (see section Data Capture and Encoding, page 24);
- User Registry management (add, modify, delete);
- Laboratory registry management (add, modify, delete);
- Internet-based administrative panel for database management (backup, restore, inventory and other database information);
- Vocabulary references and links that can be used by user interfaces and reports; and
- In similar function as Data Browse and Data View packages, data export services to CSV, TSV or XML using the data schema when appropriate.

This subtask should culminate with documentation of the package containing the technical details, user guide and parameters for proper regression testing.

10. System-level/Integration Testing and Usability

This tasks will commence with a thorough review of the functional requirements and how to test the functions. The test parameters that are provided by each of previous software development tasks are also included in the overall test program. In a workshop scenario, users will put the system to the test, and every strokes and input from the users will be recorded and available for used in regression testing.

This subtask should culminate with documentation of the package containing the technical details, user guide and guidelines for regression testing.

11. System Documentation

This task will draft a user guide and online help system. Importantly, this task will also write the final technical documentation that can be employed when updating Health MAP when the need arises. This task will culminate in the publication of three documents: user guide, technical document and online help system.

12. Production System Deployment

This task is to deploy the information system to the production server and transfer the technology to designated agencies or organizations. A series of technical training sessions is expected when transferring Health MAP for continuing management or development of the information system. This task does not end after the last technical training session. A technical help desk should persist for at least year to ensure that the system will sustained beyond a year after the transfer.

13. Trainers Training

Although Task 12 will include a series of technical training courses, these courses are designed for the technical tasks required to maintain the system. The training series on the use of Health MAP to users, including data providers, is not part of Task 12, but part of this task, Task 13. In this task, we propose users are trained not only in how to use Health MAP but also in how to train others on its effective use. The selection criteria for new trainees will be determined at the start of this task. At the end of the training course, attendees who completed the course successfully and demonstrate competence in using the system can be certified as a Health MAP Certified Trainer.

Training trainers instead of simply users will have a multiplying effect on the pool of expertise necessary to use Health MAP efficiently and help sustain the platform beyond the project development life cycle.

14. Operations and Administration

This task runs for the duration of the project. This task will coordinate all the activities of the project to ensure that deliverables and functional requirements are met, liaise with collaborating organizations, and maintain completed products or packages. The list of tasks will also include, organization of a project-wide all-hands meeting, presentation of products, and development and maintenance of project websites. The team that will be formed under this task will also house the Help Desk to address inquiries that are Health MAP-related. All user queries will be forwarded to this Help Desk.

15. Post-Implementation Review

Prior to the completion of the project, this task is to document the lessons learned when developing Health MAP. The resulting document can be used as a case study for others to reference if needed and a way to look back and assess performance.

This task will culminate with the publication of a final report including sections on lessons learned, operational deployment of Health MAP, and transfer of the technology to designated agencies or organizations.

Cost Tables

Table 15 is a cost estimate for the 3-year project (see Figure 19 and Table 16 for the Project Gantt Chart). This estimate includes the services of a lead Architect, two Software Engineers (Python), a Research Associate (MSc), a Graduate Student Assistant, and a System Administrator. Travel will include annual project meeting, field visits, and interviews. Materials and supplies are the standard office work materials. The workshops will include the OGC-lead standards development workshop in Year 1 and the project's All-Hands Meetings in Years 2 and 3 with a follow-up workshop on the OGC standards for adoption in Year 3. The servers and workstations are a cost associated with using the computers and cloud-based servers (one for production and the other as alternate and off-site backup).

Table 15. Project cost (‘000) estimate to complete the development and deployment of the proposed *Health MAP*

Item	*y1*	*y2*	*y3*	*Total*
Salary	210.3	233.0	251.2	694.6
Travel	10.0	6.0	11.0	27.0
Materials & Supplies	0.5	0.5	0.5	1.5
Workshops, Consultants	150.0	5.0	10.0	165.0
Server & Workstations	9.0	1.0	1.0	11.0
TOTAL (IDC Included)	**427.6**	**301.5**	**339.0**	**1,068.0**

Timeline, Milestones and Gantt Chart

Figure 19 is a summary of the Timeline and Milestones of the recommended tasks to develop and deploy Health MAP. The Gantt chart (Table 16) includes the details of the tasks. The resources required to complete the project in three years rests on the following assumptions:

1. The technical supervisor of the project is an architect that can lead a technical team and provide the minimum of 3-person-months per year;
2. The project starts December 2017, and two Python+PostgreSQL+XML-experienced, Software Engineers can be hired during the first quarter of the project to work full time;
3. A Research Associate with strong background in marine mammal research to assist in the day-to-day conduct of the project, documentation, training and coordination functions with stakeholders and partners, can be hired during the first quarter of the project to work full time;
4. A Graduate Assistant to assist in the operations, testing, documentation and data extraction, can be hired in the first quarter of the project; and
5. Cloud-based servers, workstations, and workspaces are available from the outset of the project.

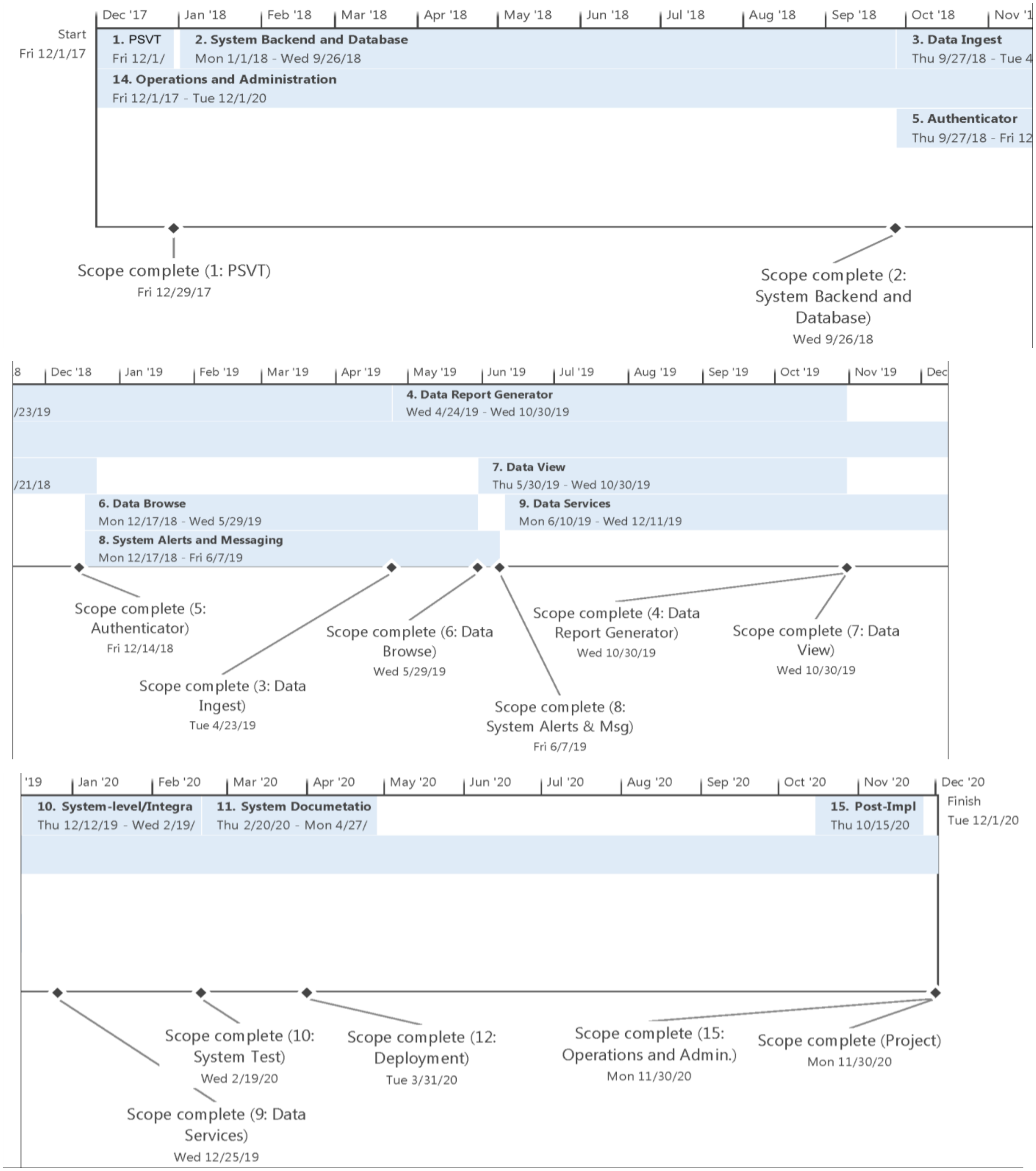
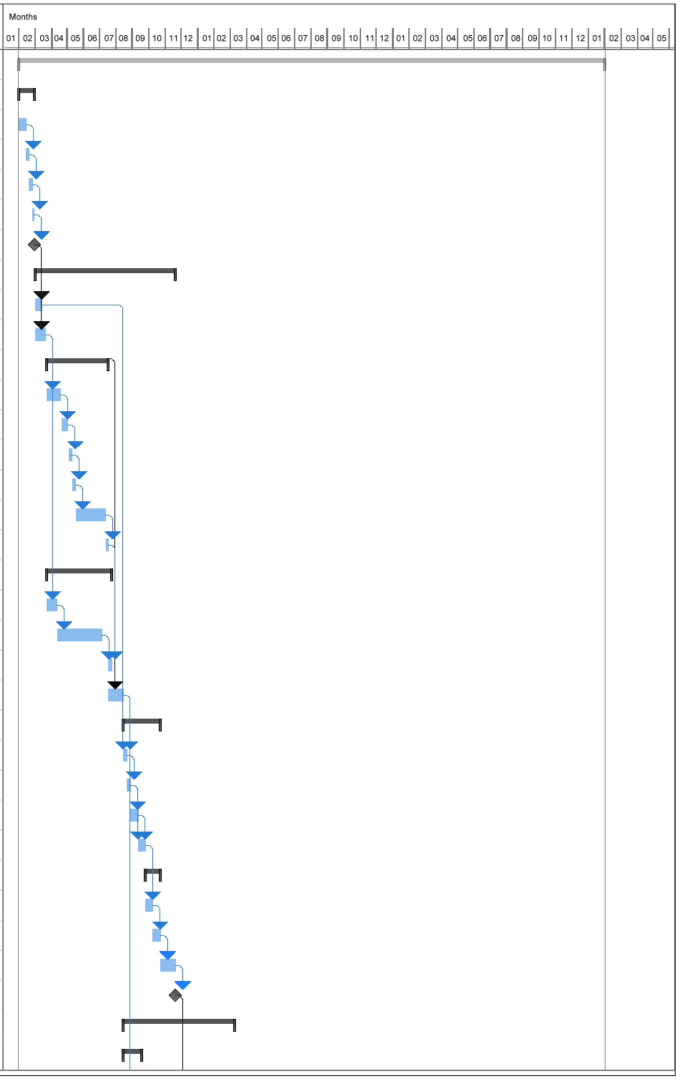


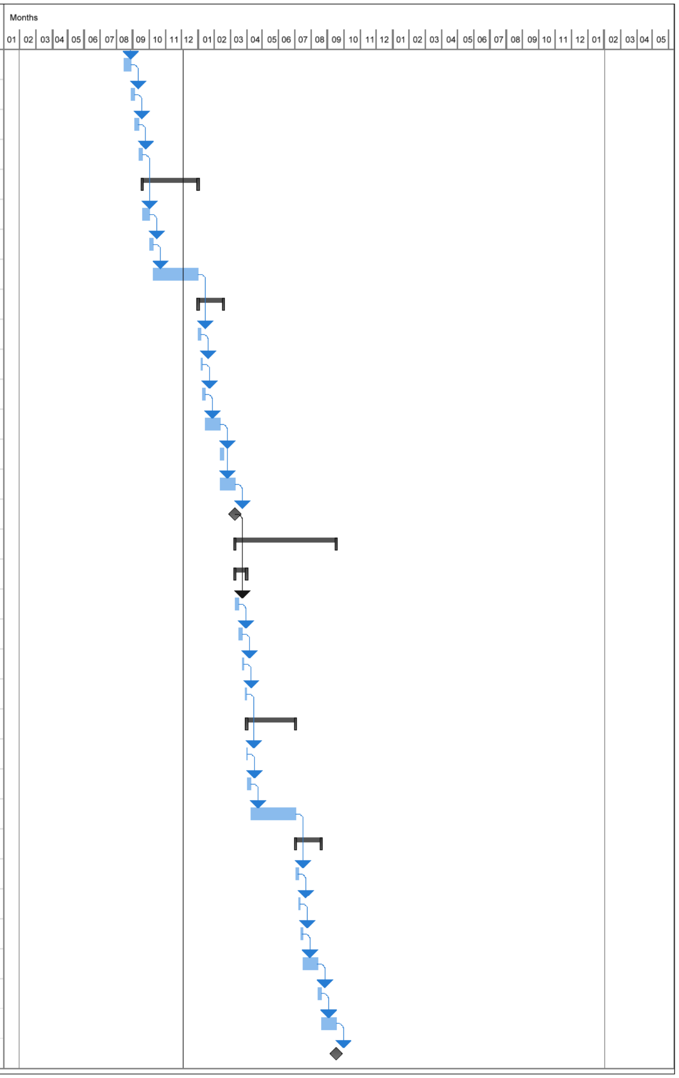
Figure 19. The proposed three-year project timeline and milestones.

Table 16. Proposed 3-year project Gantt chart. A 3-page version of this table is available at http://data.gcoos.org/mmhmap/HealthMAP_v1.pdf.

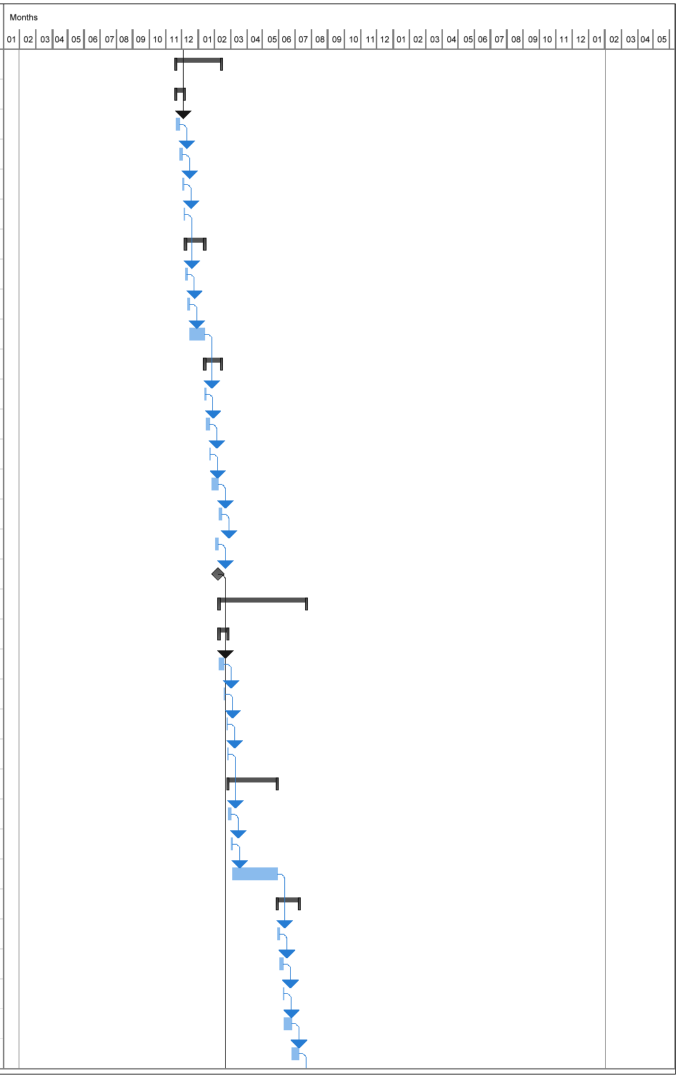
ID	Task Name
0	**HealthMAP System Development & Deployment**
1	**1. Project Scoping, Visioning, Team Building (PSVT)**
2	1.1 Pre-PSVT
3	1.2. PSVT Workshop
4	1.3. Finalize project scope
5	1.4. Secure core resources
6	Scope complete (1: PSVT)
7	**2. System Backend and Database**
8	2.1. Server initialization and installation
9	2.2. Data schema review
10	**2.3. Schema to OGC O&M**
11	2.3.1. Working Group Formation
12	2.3.2. Pre-Workshop
13	2.3.3. OGC/MMC/NOAA Workshop
14	2.3.4. Server and testbed
15	2.3.5. OGC Community review/comment
16	2.3.6. OGC Adoption
17	**2.4. Vocabulary**
18	2.4.1. Draft Table Listing
19	2.4.2. Draft Review and Concensus Building
20	2.4.3. DwC submission
21	2.5. Data Schema to DB
22	**2.6. Backup and Recovery**
23	2.6.1. Review DR requirements
24	2.6.2. Secure resources
25	2.6.3. Script for near-site backup
26	2.6.4. Script for off-site backup
27	**2.6.5. Setup alternate site**
28	2.6.5.1. Script to set mirror site
29	2.6.5.2. Script/Procedure for Domain Rollover
30	2.7. System Test and Review
31	Scope complete (2: System Backend and Database)
32	**3. Data Ingest**
33	**3.1. Review of Package Requirements**



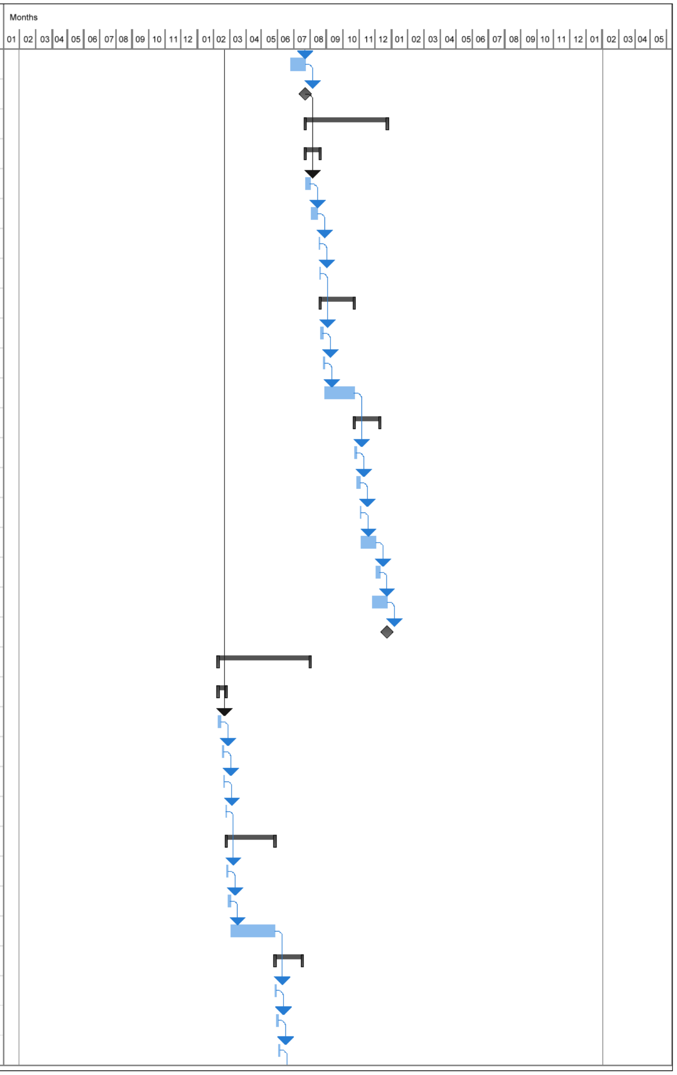
ID	Task Name
34	3.1.1. Review/conduct needs analysis & functional requirements
35	3.1.2. Finalize software specifications
36	3.1.3. Develop delivery timeline (Agile)
37	3.1.4. Secure required resources
38	**3.2. Package Development**
39	3.2.1. Review functional requirements
40	3.2.2. Identify modular/tiered requirements
41	3.2.3. Code development/test cycle
42	**3.3. Package testing**
43	3.3.1. Develop test plans based on functional requirements
44	3.3.2. Test component modules to functional requirements
45	3.3.3. Identify issues to package specifications
46	3.3.4. Update/Modify code
47	3.3.5. . Regression testing
48	3.4. Package documentation
49	Scope complete (3: Data Ingest)
50	**4. Data Report Generator**
51	**4.1. Review of Package Requirements**
52	4.1.1. Review/conduct needs analysis & functional requirements
53	4.1.2. Finalize software specifications
54	4.1.3. Develop delivery timeline (Agile)
55	4.1.4. Secure required resources
56	**4.2. Package Development**
57	4.2.1. Review functional specifications
58	4.2.2. Identify modular/tiered requirements
59	4.2.3. Code development/test cycle
60	**4.3. Package testing**
61	4.3.1. Develop test plans based on functional requirements
62	4.3.2. Test component modules to functional requirements
63	4.3.3. Identify issues to package specifications
64	4.3.4. Update/Modify code
65	4.3.5. Regression testing
66	4.4.Package documentation
67	Scope complete (4: Data Report Generator)



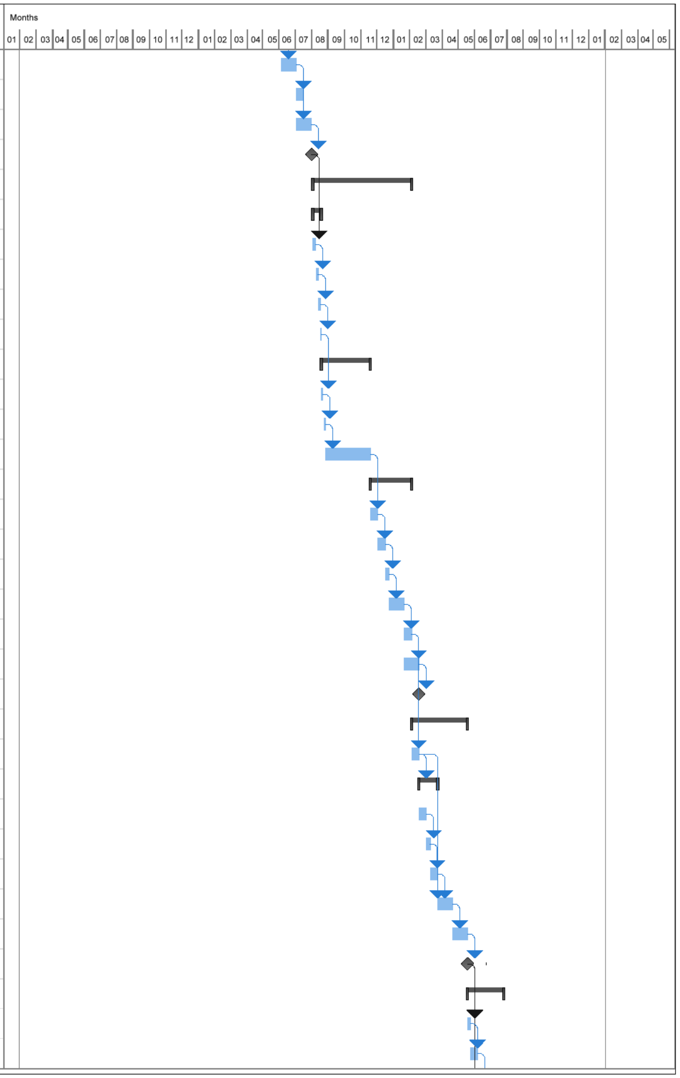
ID	Task Name
68	**5. Authenticator**
69	**5.1. Review of Package Requirements**
70	5.1.1. Review/conduct needs analysis & functional requirements
71	5.1.2. Finalize software specifications
72	5.1.3. Develop delivery timeline (Agile)
73	5.1.4. Secure required resources
74	**5.2. Package Development**
75	5.2.1. Review functional specifications
76	5.2.2. Identify modular/tiered requirements
77	5.2.3. Code development/test cycle
78	**5.3. Package testing**
79	5.3.1. Develop test plans based on functional requirements
80	5.3.2. Test component modules to functional requirements
81	5.3.3. Identify issues to package specifications
82	5.3.4. Update/Modify code
83	5.3.5. Regression testing
84	5.4. Package documentation
85	Scope complete (5: Authenticator)
86	**6. Data Browse**
87	**6.1. Review of Package Requirements**
88	6.1.1. Review/conduct needs analysis & functional requirements
89	6.1.2. Finalize software specifications
90	6.1.3. Develop delivery timeline (Agile)
91	6.1.4. Secure required resources
92	**6.2. Package Development**
93	6.2.1. Review functional specifications
94	6.2.2. Identify modular/tiered requirements
95	6.2.3. Code development/test cycle
96	**6.3. Package testing**
97	6.3.1. Develop test plans based on functional requirements
98	6.3.2. Test component modules to functional requirements
99	6.3.3. Identify issues to package specifications
100	6.3.4. Update/Modify code
101	6.3.5. Regression testing

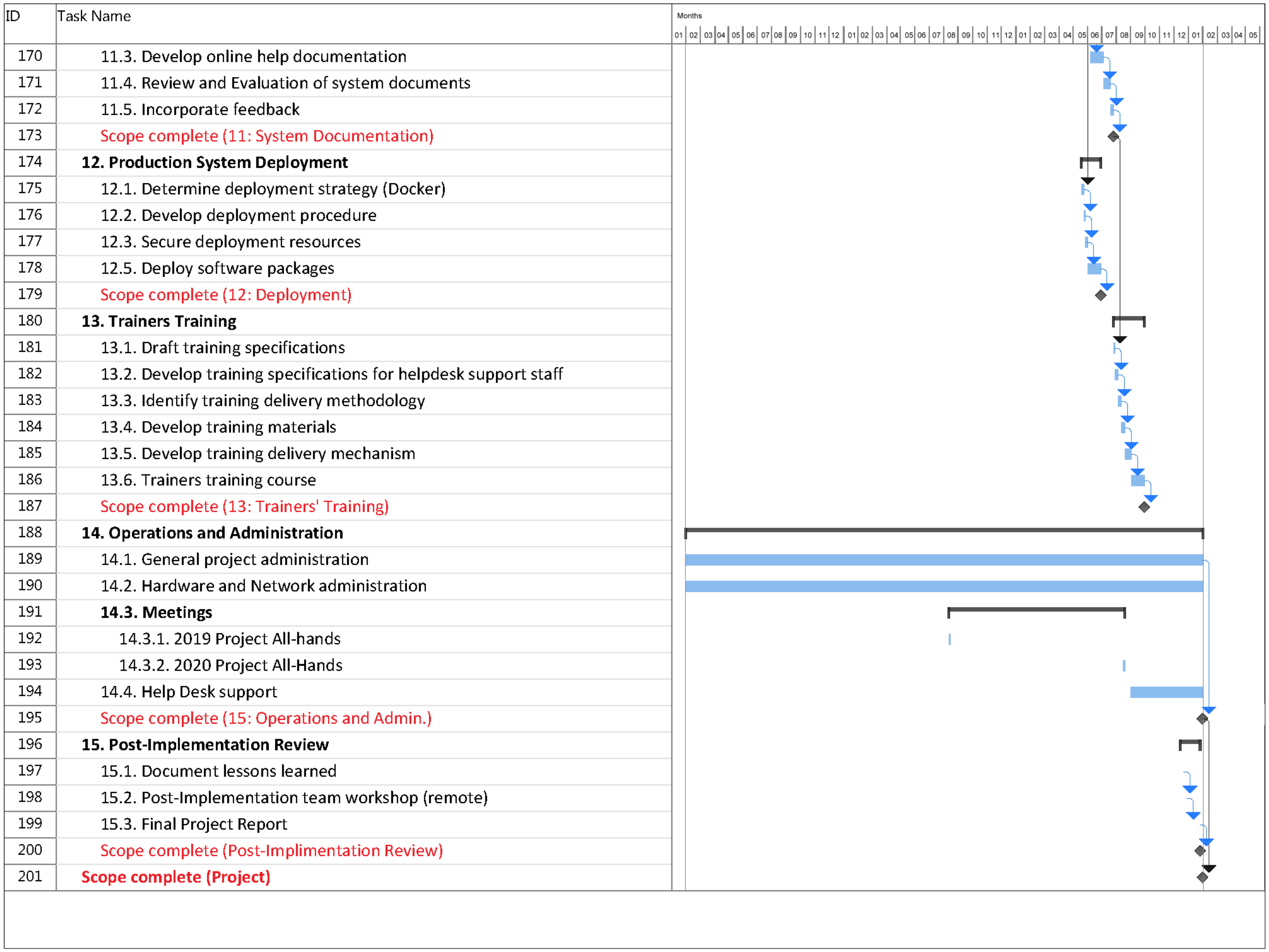


ID	Task Name
102	6.4. Package documentation
103	Scope complete (6: Data Browse)
104	**7. Data View**
105	**7.1. Review of Package Requirements**
106	7.1.1. Review/conduct needs analysis & functional requirements
107	7.1.2. Finalize software specifications
108	7.1.3. Develop delivery timeline (Agile)
109	7.1.4. Secure required resources
110	**7.2. Package Development**
111	7.2.1. Review functional specifications
112	7.2.2. Identify modular/tiered requirements
113	7.2.3. Code development/test cycle
114	**7.3. Package testing**
115	7.3.1. Develop test plans based on functional requirements
116	7.3.2. Test component modules to functional requirements
117	7.3.3. Identify issues to package specifications
118	7.3.4. Update/Modify code
119	7.3.5. Regression testing
120	7.4. Package documentation
121	Scope complete (7: Data View)
122	**8. System Alerts and Messaging**
123	**8.1. Review of Package Requirements**
124	8.1.1. Review/conduct needs analysis & functional requirements
125	8.1.2. Finalize software specifications
126	8.1.3. Develop delivery timeline (Agile)
127	8.1.4. Secure required resources
128	**8.2. Package Development**
129	8.2.1. Review functional specifications
130	8.2.2. Identify modular/tiered requirements
131	8.2.3. Code development/test cycle
132	**8.3. Package testing**
133	8.3.1. Develop test plans based on functional requirements
134	8.3.2. Test component modules to functional requirements
135	8.3.3. Identify issues to package specifications



ID	Task Name
136	8.3.4. Update/Modify code
137	8.3.5. Regression testing
138	8.4. Package documentation
139	Scope complete (8: System Alerts & Msg)
140	**9. Data Services**
141	**9.1. Review of Package Requirements**
142	9.1.1. Review/conduct needs analysis & functional requirements
143	9.1.2. Finalize software specifications
144	9.1.3. Develop delivery timeline (Agile)
145	9.1.4. Secure required resources
146	**9.2. Package Development**
147	9.2.1. Review functional specifications
148	9.2.2. Identify modular/tiered requirements
149	9.2.3. Code development/test cycle
150	9.3. Package testing
151	9.3.1. Develop test plans based on functional requirements
152	9.3.2. Test component modules to functional requirements
153	9.3.3. Identify issues to package specifications
154	9.3.4. Update/Modify code
155	9.3.5. Regression testing
156	9.4.Package documentation
157	Scope complete (9: Data Services)
158	**10. System-level/Integration Testing and Usability**
159	10.1. Test system-level/module integration
160	**10.2. Usability Workshop/Community Engagement**
161	10.2.1. Pre-workshop activities
162	10.2.2. Workshop proper
163	10.2.3. Post-workshop
164	10.3. Modify code (assumed necessary)
165	10.4. Regression testing
166	Scope complete (10: System Test & Usability)
167	**11. System Documentation**
168	11.1. Help system specification
169	11.2. Evaluate online help and drafted materials





ID	Task Name
170	11.3. Develop online help documentation
171	11.4. Review and Evaluation of system documents
172	11.5. Incorporate feedback
173	Scope complete (11: System Documentation)
174	**12. Production System Deployment**
175	12.1. Determine deployment strategy (Docker)
176	12.2. Develop deployment procedure
177	12.3. Secure deployment resources
178	12.5. Deploy software packages
179	Scope complete (12: Deployment)
180	**13. Trainers Training**
181	13.1. Draft training specifications
182	13.2. Develop training specifications for helpdesk support staff
183	13.3. Identify training delivery methodology
184	13.4. Develop training materials
185	13.5. Develop training delivery mechanism
186	13.6. Trainers training course
187	Scope complete (13: Trainers' Training)
188	**14. Operations and Administration**
189	14.1. General project administration
190	14.2. Hardware and Network administration
191	**14.3. Meetings**
192	14.3.1. 2019 Project All-hands
193	14.3.2. 2020 Project All-Hands
194	14.4. Help Desk support
195	Scope complete (15: Operations and Admin.)
196	**15. Post-Implementation Review**
197	15.1. Document lessons learned
198	15.2. Post-Implementation team workshop (remote)
199	15.3. Final Project Report
200	Scope complete (Post-Implimentation Review)
201	**Scope complete (Project)**

Modularized Development

Given the modularity of the design, the project can be completed in less than three years if several contractors can be engaged at the same time. The following describes a task implementation schedules if the project is broken down into smaller components. The increased cost estimate is associated in large part with the higher than regular hourly rates of workers given the shorter period of employment. The other items that lead to an increase in the cost of the product include:

- Project management and coordination tasks;
- Training and technology transfer;
- Functional reviews and assessments; and
- Modularize testing and issue tracking.

Package 1: Primary and essential components

This package will include pre-requisite modules and submodules from which all the other modules will be based. The essential components will include the development of:

- Data Capture: Basic user interfaces to capture data from desktop and laptop computers;
- Data Browse: Simple table presentation of the captured data with faceted search functions;
- Data View: User interactive map of some attributes of the data in the database
- Data Services: Development of the Persistent Identity Generator (idGen), User Registry and Laboratory Registry;
- Data Report Generator: Predefined report generator for application with Data Browse, Data View, and Data Services.
- Data Authenticator: Authentication only via Google OpenID Provider
- Health MAP Code DB; and
- User Registry and Laboratory Registry DBs.

This package will also include the following tasks:

- Deployment of the information system in a cloud server;
- Maintenance of the information system for three years;
- Transfer of the technology at the end of the 3rd year to designated agency or organization;
- Integrated system-level testing of the modules;
- Documentation of the completed products (technical, user guide and testing parameters); and
- Training of selected users.

The following is the cost estimate for Package 1.

Item	*Value*
Pre-requisite	None
Duration	18 months
Approximate Cost	$250,000.00

Package 2: Standards development

This package will include tasks associated with the development and registration of a common vocabulary, and activities to extend the Darwin Core with the Health MAP terms and definitions. It will commence with review of the draft collection of terms and definitions, mapping with DwC and submitting the terms for DwC consideration.

The proposed data schema will be examined, and OGC O&M extended to ensure that the proposed Health MAP parameters are completely represented. A test server will be installed to test query the system. MongoDB is the preferred No-SQL-based system to be used. The proposed extension will be submitted to OGC for adoption.

Item	*Value*
Pre-requisite	None
Duration	6 months (0.2 months in year 3)
Approximate Cost	$200,000.00

Package 3: Data Ingest Advanced Features

Advanced features not included in the Data Ingest core components developed in Package 1 will include:

- Mobile apps for data capture and online or delayed (absence of connectivity) reporting;
- Capture of Mobile GPS information to facilitate coordinate recording;
- Capture and storage of images directly to the system and mobile device enabled; and
- Nomenclature validation with an external source.

Item	*Value*
Pre-requisite	Package 1
Duration	10 months
Approximate Cost	$100,000.00

Package 4: Data View Advanced Features

Advanced features not included in the Data View core components developed in Package 1 will include:

- Overlay of other base maps, ecological and environmental data layers that are available from external sources;
- Spatial-based query and geostatistical analyses;
- Customized query, storage of these queries, and retrieval of stored queries; and

- 2D/3D graphical presentation of the queried spatial data when applicable.

Item	*Value*
Pre-requisite	Package 1
Duration	10 months
Approximate Cost	$100,000.00

Package 5: Data Browse Advanced Features

Advanced features not included in the Data Browse core components developed in Package 1 will include:

- User-interactive and customizable table structure;
- Export of results to another data type (CSV, TSV, and XML); and
- 2D/3D graphical presentation of the queried spatial data when applicable.

Item	*Value*
Pre-requisite	Package 1
Duration	10 months
Approximate Cost	$50,000.00

Package 6: Data Services Advanced Features

Advanced features not included in the Data Services core components developed in Package 1 will include:

- Extends the ID generator to map with other identifiers when available;
- Provide an interface to present other details about a marine mammal extracted from external sources; and
- Direct access to the National Stranding Database to retrieve data and plot when applicable.

Item	*Value*
Pre-requisite	Package 1
Duration	6 months
Approximate Cost	$80,000.00

Package 7: Usability Testing and System Assessment

This package will focus on the usability and evaluation of the system through usability workshops. The final product, as stipulated above, will be recommendations to improve the initial release of the information system (features and functions).

Item	*Value*
Pre-requisite	Packages 1, 2, 3,4 ,5, and 6
Duration	4 months
Approximate Cost	$50,000.00

Package 8: Disaster Recovery and Risk Mitigation

This package will focus on the "what-if" scenario to ensure high availability, security, and data access controls. This package will include extending the Authenticator package developed in Package 1 to add other identity providers. In addition, to address all other issues listed in Task 2.6 (Backup and Recovery).

Item	*Value*
Pre-requisite	Package 1
Duration	4 months
Approximate Cost	$50,000.00

Package 9: System Alerts and Messaging

This task is to implement Task 8, i.e., establish the messaging system, which includes the development of mobile device applications to handle system messages.

Item	*Value*
Pre-requisite	Package 1
Duration	10 months
Approximate Cost	$200,000.00

Package 10: Trainers Training

This task is Task 13 as described above, i.e., to develop an advanced group of users with the capability to train others to maintain the pool of expertise needed to sustain the application of the information beyond the project life.

Item	*Value*
Pre-requisite	Package 1
Duration	4 months
Approximate Cost	$60,000.00

Package 11: System Management and Maintenance

This task is associated with Task 14 (Operations and Maintenance) to assist with the coordination of the various tasks, and the organization of Health MAP related activities, including the planning and execution of workshops.

Item	*Value*
Pre-requisite	Package 1, Package 8
Duration	36 months
Approximate Cost	$160,000.00

Package-Based Cost Summary

The following is a summary of the costs associated with modularized development of Health MAP.

Package	***Label***	***Duration (months)***	***Approx. Costs ($)***
Package 1	Primary & Basic Components	18 months	450,000.00
Package 2	Standards Development	6 months	250,000.00
Package 3	Data Ingest Advanced Features	10 months	200,000.00
Package 4	Data View Advance Features	10 months	200,000.00
Package 5	Data Browse Advanced Features	10 months	150,000.00
Package 6	Data Services Advance Features	6 months	150,000.00
Package 7	Usability and Assessment	4 months	150,000.00
Package 8	Disaster Recovery & Risk Mitigation	4 months	200,000.00
Package 9	System Alerts and Messaging	12 months	200,000.00
Package 10	Trainers Training	4 months	60,000.00
Package 11	System Management & Maintenance	36 months	160,000.00
TOTAL			**$1,300,000.00**

References

[1] Wildlife Health Information Sharing Partnership (WHISPers). National Wildlife Health Center, USGS. https://www.nwhc.usgs.gov/whispers/

[2] ANSI Standards. https://docs.oracle.com/cloud/latest/db112/SQLRF/ap_standard_sql001.htm#SQLRF55514

[3] PostgreSQL. https://www.postgresql.org/

[4] MariaDB. https://mariadb.org/

[5] CentOS. https://www.centos.org/

[6] OpenID. http://openid.net/

[7] InCommon. https://www.incommon.org/

[8] Shibboleth. https://shibboleth.net/

[9] CILogon. http://www.cilogon.org/

[10] RabbitMQ. https://www.rabbitmq.com/

[11] NOAA, Marine Mammal Health and Stranding Response Program. https://mmhsrp.nmfs.noaa.gov/mmhsrp/

[12] World Registry of Marine Species (WoRMS). http://www.marinespecies.org/

[13] NOAA, Data Integration, Visualization, Exploration, and Reporting (DIVER) https://www.diver.orr.noaa.gov/

[14] Digital identity guidelines. 2017. National Institute of Standards and Technology (NIST). https://pages.nist.gov/800-63-3/sp800-63-3.html

[15] Pentaho Integration Platform. http://www.pentaho.com/product/data-integration

[16] L-Soft LISTSERVE. http://www.lsoft.com/

[17] Digital Object Identifier (DOI). http://www.doi.org/

[18] DataCite. https://www.datacite.org/

[19] AWStats. http://www.awstats.org/

[20] PiWIK Open Analytics Platform. https://piwik.org/

[21] Google Analytics. https://analytics.google.com

[22] Olea, R.A. Optimal Contour Mapping Using Universal Kriging. 1974. Journal of Geophysical Res. 79:5. 695-707p.

[23] Uptime Robot. https://uptimerobot.com/

[24] Nagios. https://www.nagios.org/

[25] *Moats, Ryan (M* (Moats, 1997)*ay 1997).*"Request for Comments: 2141: URN Syntax". *IETF. Retrieved 2012-12-07.*

[26] Internet Assigned Numbers Authority. https://www.iana.org/

[27] *Internet Corporation for Assigned Names and Numbers*. https://www.icann.org/

[28] Darwin Core RDF. (http://rs.tdwg.org/ dwc/terms/guides/rdf/)