NERACOOS Data Framework Plan
Developed by the Gulf of Maine Research Institute, NERACOOS DMAC provider
Latest update: 10/10/14

I. Purpose, Scope and Objectives

The purpose of this document is to describe the NERACOOS data management framework and implementation plan for the various data types produced by NERACOOS and key partners. The plan outlines how:

1. Funded partners should make their data discoverable and accessible using IOOS recommended standards and services.
2. External partners data is accessed or can be made accessible
3. An implementation plan for bringing data providers into compliance with the framework

The DMAC framework serves as the foundation to the NERACOOS data portal and all products and services that rely on these data. The framework leverages the suite of standards adopted and recommended by IOOS. The NERACOOS DMAC framework will be the structure behind the goal of having a centralized source for regional data access, and in some cases, storage.

Because many partners are currently capable of storing and serving data in standard compliant formats (e.g. NetCDF) that can be ingested by the NERACOOS data center, the goal is not to aggregate these data in a centralized warehouse. Guidance is provided to NERACOOS data providers so they can output their data in compliant formats to avoid duplication from additional storage where not necessary. However, for data providers lacking bandwidth or capacity to serve data reliably or in compliant formats, the framework has the capacity to harvest, store and serve these data in standardized output formats.

The NERACOOS Data Portal, described in separate documentation, will provide the front end for discovery and paths to access the historical and real-time data from these distributed systems.

I. Resources and Definitions:

1. Funded Partners

The NERACOOS observations include real-time data from buoys and sensors, model outputs and satellite

Observations

- UMaine - Real Time/Historic
- UConn - Real Time/Historic
- URI - Real Time/Historic
- UNH - Real Time/Historic

- High Frequency Radar (HFR)

Models

- University of Massachusetts Dartmouth - NECOFS
    - Parameters: currents, water level, temp, salinity, waves-hindcast/forecast
- Bedford Institute of Oceanography - WW3
    - Parameters: waves (forecast/hindcast)
- UConn
    - HFR STPS (nowcast/forecast)

Satellites

- UMaine

Other NERACOOS Partner Data Providers (non PI)

- Observations:
    - eMOLT
    - ME DMR
    - NERRS
- Models
    - UMaine GoM Circulation model includes currents, temp and salinity
- Satellites
    - NOAA Coastwatch

External Partners - Phase 2

Data from external partners could include USGS tide gauge, NWLON, NERRS, NDBC, NEPs, Exchange Network, gliders, ship data, NEFSC drifters, eMOLT, NEXRAD, and others TBD.

Need to clearly define and prioritize this group and goals for providing access to the data.

2. Data Types

- Priority: station data (buoys, tide gauge, etc.) models
    - Data readiness criteria:
        - What format and version is data in from data provider (NetCDF, other)?
        - Is provider serving via THREDDS server already?
- Secondary: satellite, HFR, glider, ship, etc
    - Additional criteria to be determined for secondary data:
        - Are there different standards/services beyond SOS or THREDDS that are needed to handle HFR/glider/ship data?

                - Is NetCDF common for these data types? Or is Satellite using HDF?
                - Define the specific entities/providers that are serving glider/satellite/ship data to scope and prioritize

3. Standards and Services

- Using IOOS Recommended OGC Standards
- New IOOS SOS SWE Profile
    - OGC SWE, CF, Vocabularies at MMI, mapped SOS to UCAR NetCDF Common Data Model, SensorML metadata
    - Discrete Sampling Geometries implemented in the NetCDF Java library (described in IOOS Modeling Testbed project)

- Leverage IOOS Funded Tools to assist RAs
    - Java based ncSOS
    - 52North SOS
    - JS / Python Clients in development

- SOA and Data Aggregation Center
    - SOA - Data Providers should control their data and metadata
    - Still a role for DAC

II. Data Provider Requirements

1. Requirements for NERACOOS Data

Goal: Observation data are output in standard, accessible format

a. Data format: NetCDF will be the basic data format supported for data output. Most observation data providers already use NetCDF as their data storage format, or are capable of producing NetCDF, and the software implemented in the framework is capable of processing data in this format.

b. Metadata: CF 1.6, the current version of the Climate Forecast convention, will be required either in the NetCDF Headers or via THREDDS (TDS) ncML plugin.

    i. The protocol to update via ncML has been well documented and is typically set up once per provider and updates aren't necessary unless major changes happen with the data collection protocols. At that time, the headers will need to be updated. Middleware is available to to allow DPs to update metadata through ncML (Signell 2014).

c. Unidata's Common Data Model will be utilized. In particular the Discrete Sampling Geometries for Point Time Series.

2. **Requirements for External Data**

a. Need to more clearly define universe external data partners.
b. Understand existing data access formats for the various partners

III. Structure of NERACOOS Data Framework

1. **Storage**
    - **a.** Amazon EC2 (Elastic Cloud Computing)
        - **i.** Cloud services provide cost savings, scalability and shared resources, though it should be noted, increased data serving will likely cause increased monthly costs associated with hosting and bandwidth to access.
            - **1.** Initial deployment of the framework is on the existing EC2 instance. The monthly data use will be monitored during the first year of operation and if deemed necessary, may result in setting up a separate instance dedicated to data storage and access.
        - **ii.** GMRI manages the cloud computing instance for NERACOOS and recommend this solution for the data framework and associated data access products.
2. **Data formats**
    - **a.** At the core of the NERACOOS Data Framework is THREDDS Data Server (TDS): Thematic Real-Time Environmental Distributed Data Services.
        - **i.** TDS plugin architecture utilizes a suite of standards that handle in-situ obs and models with same protocols.
        - **ii.** A TDS is installed on the NERACOOS cloud environment.
        - **iii.** Data is accessed via:
            - **1.** direct communication with data partners through TDS to TDS communication (UMass Dartmouth, UConn)
            - **2.** Or by accessing NetCDF files via remote directories (UMaine, UNH, or actual files on NERACOOS servers).
        - **iv.** TDS will be utilized to serve the NERACOOS provider data in IOOS DMAC compliant formats:
            - 1. WMS, SOS, ISO 19113 Metadata, OpenDAP (access to NetCDF or subsets), etc.
            - 2. Benefits:
                - a. Data providers control data and metadata
                - b. Post Recovery files QA/QC replace provisional real-time as available, TDS/ERDDAP recalibrate for new data automatically
                - c. Creates SOS service for single parameter requests or time-series requests for data products
        - **v.** TDS can make collections of single files (i.e. all deployments of one buoy that may be in individual files)

        vi. Various TDS plugins have been installed to accomplish this:
            - ncML, ncISO, ncWMS, ncSOS. OPeNDAP access is standard in TDS.

1. **Catalogs, Metadata, Vocabularies**

    a. The NERACOOS framework has and will continue to leverage existing work at national level with regard to cataloging and metadata and vocabulary standards

        i. Utilize ISO 19115 Metadata Standard
        ii. Contribute to IOOS catalog
        iii. Develop localized NERACOOS catalog

    b. Utilize ncML to make data CF and ISO compliant

        i. CF common name in ncML
        ii. Proper metadata (ISO compliant; IOOS cert requirement)

1. **QA/QC**

    a. Recommendations and guidance provided by IOOS QARTOD effort will be followed.

    b. The framework allows data providers to remain the direct source for highest QA/QC NetCDF files, enabling post recovery recalibration of data and an automated mechanism for updating these files.

2. **Data sharing/delivery**

    a. An ERDDAP server, a human readable interface to TDS, has been set up and will be used as front end for speedy data access in numerous formats.

        i. ERDDAP uses NetCDF as a data source
        ii. Produces files in common outputs: GeoJSON, HTML, CSV, REST API, etc.
        iii. Will serve as front end for advanced user, and backend for general user data access (via Graphing and Download Tool)

3. **Data products**

    a. Historic Data Access Portal

        i. Update of the existing Graphing and Download Tool

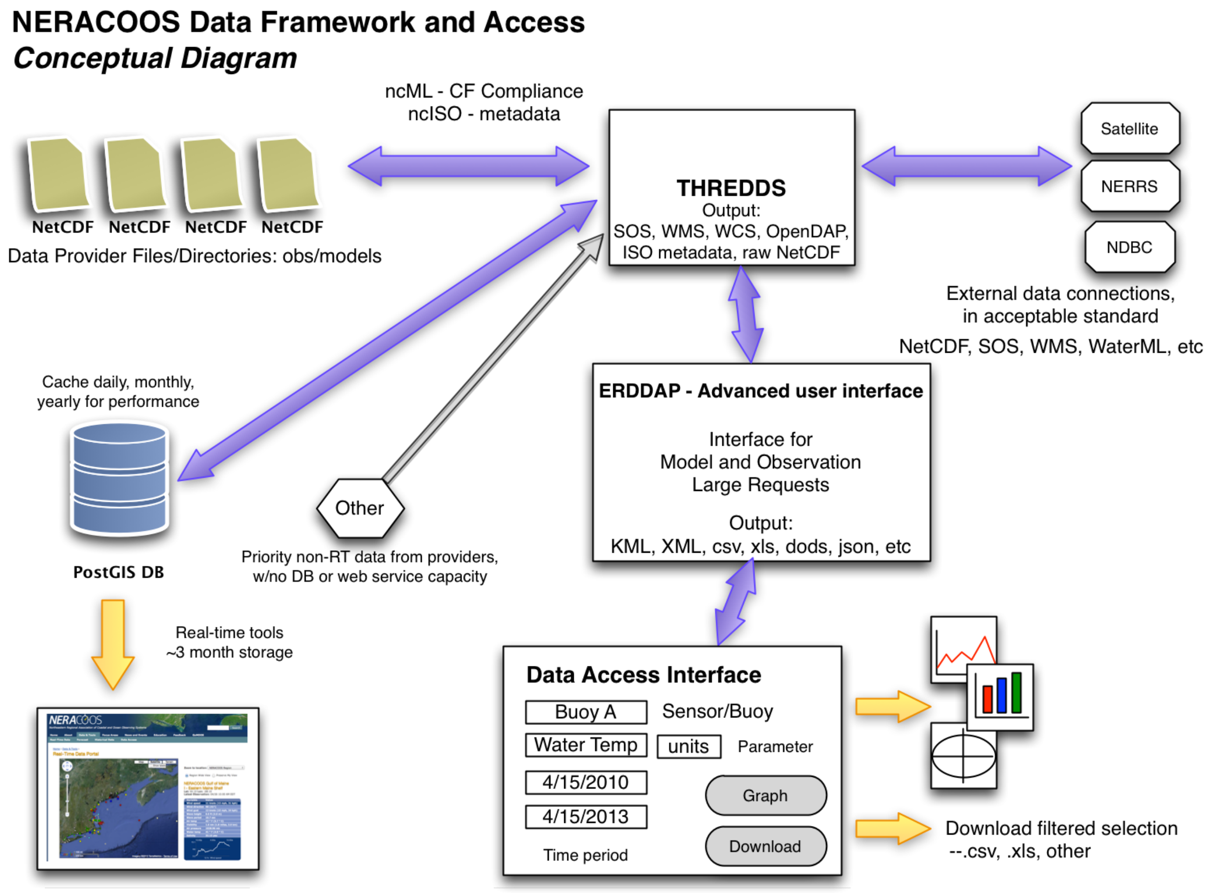

Figure 1: Framework Concept Diagram - this diagram demonstrates the configuration of the NERACOOS Data Framework, from data ingest, to processing, to output.

IV. Implementation Plan

The primary goal of the NERACOOS Data Framework is to ensure that the highest QA/QC data (historical and real-time) is available in standardized format for all partners and accessible for products and services (e.g. NERACOOS Data Portal). In future iterations, the framework will be leveraged to ensure external partner data is discoverable and accessible through products and services.

The basic requirement is that that data is available in NetCDF or via TDS connection with appropriate metadata. For data providers not currently serving NetCDF, efforts were undertaken to understand capacity and potential for serving NetCDF (i.e. Python scripts to translate text to NetCDF). Metadata records were reviewed with data providers and guidance given on how to update records with missing information.

Phase 1: NERACOOS Funded Partners

- At the SPI meeting in April 2013, the draft framework and roll out plan was introduced to all partners
- Initial one-on-one meetings were held with each data provider to understand road to implementation for each partner and potential hurdles or issues and time/level of support needed
    - Questions included:
        - What is format of existing data output? NetCDF? Database? Other
        - What is frequency/volume of data?
        - What is metadata status?
            - Use existing scripts/rubrics to determine metadata status/score
            - Help data providers contribute missing information
        - Does DP have or have capacity/interest to host own THREDDS server?
            - More likely the case with model data providers (e.g. UMass)
- Working with partners, process will be developed for maintaining and evolving framework as new standards are developed, this includes a path to upgrade for data partners.

Phase 2: External Partners - an outline of this effort as funding and time allow

- Quantify universe of external partners
- Identify and assess available formats for accessing data currently, and level of participation expected
- Understand size of data holdings
- Understand and document use case of data -
    - How far back are we going for historical data from external partners?
    - Is the expectation that all data will it be available via NERACOOS data portal?
    - Or is our goal to make it easier to access directly from provider by providing endpoints and catalog of services?
- Assume similar protocol as outlined for partners for implementation once defined

Technical implementation (Back end system development)

<u>Real-time and Historic Observations via NetCDF / TDS / ncSOS</u>

1. Created AWS Ubuntu 10.04.2 LTS instance

    a. Installed Apache 2.0, Tomcat 7, Java SDK 1.7, Java NetCDF 4.3, THREDDS 4.3
    b. Installed TDS plugins including: ncISO, ncML, ncWMS, etc
    c. Configured TDS with default data sets
    d. Confirmed operation
    e. Documented process
2. Installed / configured Python 2.7 Scientific Development Framework
3. Created binary AWS Instance and saved copy to use for restart, sharing, etc
4. Rsync-ed UMaine NetCDF files for A01 as first test case
    a. Configured TDS to serve these via standard OPeNDAP filter form and HTTP access
    b. Configured TDS ncML to add CF Metadata to A01 sensors
5. Installed and configured ncSOS
    a. Worked with ASA and PacIOOS leveraging their past and successful usage of NetCDF for time series observation via ncSOS
    b. Tested SOS with Python client parsers
    c. Replicated with remaining UMaine Buoys
    d. Utilized TDS ncML files to make files CF 1.6 compliant
    e. Developed ncML examples and guidelines for other Data Providers
6. Configured ncML to ensure ncISO output is sufficient for IOOS Catalog and Asset Inventory
7. Added remote connection to UConn TDS server to framework
8. Added UNH data to framework
9. Registered data with IOOS Catalog

Models NECOFS FVCOM/WW3

1. NECOFS
    a. Connected to SMAST TDS via remote access
        i. FVCOM is an irregular grid, efforts to process via ncSOS are underway with IOOS Model Testbed project
    b. Installed python SciWMS (ASA) for visualization of FVCOM NetCDF
    c. Developed generic python script to extract FVCOM timeseries via lat,lon
2. WW3
    a. Accessed WW3 via remote access to WHOI TDS
        i. WW3 is a regular grid and ncSOS can handle this.
        ii. Leveraged efforts with GLOS FVCOM.

Notes: This approach relies on accessing model output directly from data providers. Hosting or rsyncing large NetCDF files from models/satellites will likely require an increase in bandwidth costs that may be out of budget/scope for NERACOOS.

ERDDAP

ERDDAP provides a human readable user interface for accessing NetCDF data sets in numerous formats and will serve as an "Advanced User" interface for data access in a variety of formats. The process for set up and configuration is well documented (http://coastwatch.pfeg.noaa.gov/erddap/download/setup.html)

1. Installed and configured under tomcat on NERACOOS AWS EC2 instance
2. Configured server with support from ERDDAP developer Bob Simons

Archiving at NODC

Upon full implementation of the framework and full access to partner data, efforts to archive data at the NODC will be pursued. From initial information gathering with NODC the following steps will be followed:

1. Establish agreement with NODC
2. Evaluate and complete NODC's NetCDF templates
3. If transforms are necessary, leverage ncML or python tools to make data format acceptable to NODC
4. Develop automated processes for archiving of UMaine, UConn NetCDF files

Software/Hardware

The software stack indicated below has been implemented as the basic machine setup and is saved as an AWS Instance which can be shared as a binary snapshot for easy setup.

- Apache 2.0
- Apache Tomcat 7
- Java JDK 1.7
- Java NetCDF / CDM Library 4.3.15
- THREDDS Data Server 4.3.15
  - This version of Java NetCDF and TDS provides integrated ncML, ncWMS and ncISO support. Also integrated is OPeNDAP, HTTP file access and ncWCS services.
- NetCDF Tools UI Java application
- ncSOS plugin being developed by ASA (*important to note it is still in development)
- Python toolkit (not strictly part of the DMAC Framework)
  - Python 2.7, brewmaster, pip, etc.
  - SciPy, pyNetCDF4, iPython, lxml, owslib, Numpy, etc.
  - SciWMS from ASA for FVCOM irregular gridded models
- Other Linux tools: rsync, ftp, ssh, etc.
- ERDDAP

Initially, the stack was run as a separate instance for framework stack outside of NERACOOS website/DB for testing and evaluation. Upon completion, the stack was migrated to the NERACOOS AWS production website. A separate instance for the data framework toolkit is a possibility if bandwidth begins to interfere with day to day operations of the website. Potential hosting costs beyond production server and database will be researched. The need for a separate instance will be reevaluated after the first year.

V. Risk reduction in case of disasters/extreme events

The goal is to keep critical data services available during disasters or extreme events
NERACOOS partner data. Work to improve the capacity to harden data flows is the primary compenent of the Sandy Supplemental effort funded in 2014.

- Buoy data
    - Goal: Real-time data accessible during power outages or inaccessibility
    - Process:
        - DMAC team working with data providers to determine capacity to use existing or on-demand cloud or remote server instances for direct buoy communication (cell/goes), processing and outputting NetCDF files.
    - Outcomes:
        - Develop process for NetCDF files updated upon return of services via data logger
        - Develop a cost estimate for cloud space and time needed for DP to set up protocols
- Models
    - Goal: Latest output accessible via THREDDS before shutdown
    - Process:
        - UMass Dartmouth and Bedford Institute of Oceanography are working with ASA to migrate model to the cloud so critical elements can run during storm events that will shut servers down
    - Outcomes:
        - Models continue to run on remote/distributed systems during storm events
        - Automated process for fail over and synchronization after storm event
        - Cost estimates for on-demand cloud instance utilization will be developed and provided to NERACOOS

Glossary of Terms

AWS - Amazon Web Services
EC2 - Amazon Elastic Cloud Compute a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.
CF - Climate and Forecast, conventions of the NetCDF file format

DAC - Data Assembly Center, NOAA's National Buoy Data Center (NDBC) functions as the DAC for NOAA IOOS data.
DMAC - Data Management and Communications
ERDDAP - a data server that provides a simple and consistent way to plot and download subsets of gridded and tabular scientific datasets in common file formats as well as generate plots and maps.
FVCOM - Finite Volume Coastal Ocean Model
MMI - Marine Metadata Interoperability Project, a project to support collaborative research in the marine science domain, by simplifying the incredibly complex world of metadata into specific, straightforward guidance.
ncISO - A command-line utility for automating metadata analysis and ISO metadata generation for THREDDS Catalogs
ncML - NetCDF markup language
ncSOS - a service developed by Applied Science Associates. NcSOS adds an OGC SOS service to datasets in your existing THREDDS server. It complies with the IOOS SWE Milestone 1.0 templates and requires your datasets be in any of the CF 1.6 Discrete Sampling Geometries.
NDBC - NOAA's National Data Buoy Center.
NetCDF - Network Common Data Format
NODC - NOAA's National Oceanographic Data Center
OGC - Open Geospatial Consortium
OpenDAP - an acronym for "**Open-source Project for a Network Data Access Protocol**", is a data transport architecture and protocol widely used by earth scientists.
PostGIS - an open source software program that adds support for geographic objects to the PostgreSQL object-relational database.
PostgreSQL -
SciWMS
SensorML - SensorML provides standard models and an XML encoding for describing sensors and measurement processes.
SOA - Service Oriented Architecture
SOS - an OGC standard applicable to use cases in which sensor data needs to be managed in an interoperable way. This standard defines a Web service interface which allows querying observations, sensor metadata, as well as representations of observed features.
SWE - The OGC's Sensor Web Enablement (SWE) standards enable developers to make all types of sensors, transducers and sensor data repositories discoverable, accessible and useable via the Web.
TDS - THREDDS Data Server
THREDDS - Thematic Real-time Environmental Data Distributed Services
WW3