**Employing Cyber Infrastructure Data Technologies to Facilitate IEA for Climate Impacts in NE & CA LME's (ECO-OP)**

**Principal Investigator:** Peter Fox and Andy Maffei

**Sponsor:** National Science Foundation

**Description:** The purpose of this INTEROP proposal is to facilitate the deployment of an Integrated Ecosystem Approach (IEA) to management in the Northeast and California Current Large Marine Ecosystems (LMEs). The direct result of the proposed activity will be application-level data and information enhanced communication for developing the consensus networks to define the specific components of interest to support the implementation of NOAA’s Driver-Pressure-State-Impact Response framework (DPSIR) decision framework and the cyberinfrastructure technologies to ensure data interoperability and reuse.
Three Tiers

1. Support for Ecosystem Status Report
2. Support for Fishery Ecosystem Plan
3. Support for Integrated Ecosystem Assessment
Three Tiers

1. Support for Ecosystem Status Report
2. Support for Fishery Ecosystem Plan
3. Support for Integrated Ecosystem Assessment

Working with smaller group on topic 1

Purpose here is to present status and discuss next steps
1) Introduction to the ESR Use Case (Jon)
2) Software demo video
3) Discuss 4 types of software users (vs. 2 in demo video)
4) Interactive IPython demo (Massimo)
5) "Data-to-indicator-pipeline" diagram (provenance of data)
6) Feedback and next steps
Support for Ecosystem Status Report

Current Model

Individual EcoAp Scientists

- Data Source
- Data Prep
- Data Analysis
- Data Graphics
- Data Interp
- ESR

Need to move to:

- Traceability, repeatability, explanation, verification, and validation

- Have these issues individually but not from the perspective of the group or from users of the ESR.
Support for Ecosystem Status Report

Moving Forward

Use Scientists for Science

Use Computers for Routine Actions

The ESR Software App supports the routine “collection”, preparation, analysis and graphing of indicators in the ESR that also enhances *Traceability, repeatability, explanation, verification, and validation.*
Use Case
created in Workshop with EcoAp and ECOOP Scientists
Use Case
made official by ECOOP Scientists
Scoping out the Ecosystem Status Report we envision 4 types of users:

1. General - through pdf
2. EcoAp - through pdf or GUI
4. System Admin
pdf functionality

http://ligo.tw.rpi.edu:8888/

Driver test
Play Video

http://epifanio.whoi.edu/esr/ECODEMO4.swf

Go to desktop
Python r graphics

http://ligo.tw.rpi.edu:8888/

ESR_R_phisical
Combination (Satellite, Climate, GUI)

http://ligo.tw.rpi.edu:8888/

ICES Notebook
<table>
<thead>
<tr>
<th>Data and indicators coded for ESR</th>
<th>Public URL or private dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Climate Forcing</strong></td>
<td></td>
</tr>
<tr>
<td>NAO</td>
<td>public</td>
</tr>
<tr>
<td>AMO</td>
<td>public</td>
</tr>
<tr>
<td>NIN</td>
<td>public</td>
</tr>
<tr>
<td><strong>Physical Pressures</strong></td>
<td></td>
</tr>
<tr>
<td>Gulf Stream Location</td>
<td>public</td>
</tr>
<tr>
<td>FL Current Transport</td>
<td>public</td>
</tr>
<tr>
<td>River Flow GOM</td>
<td>public</td>
</tr>
<tr>
<td>River Flow MAB</td>
<td>public</td>
</tr>
<tr>
<td>River Flow SNE</td>
<td>public</td>
</tr>
<tr>
<td>ERSST</td>
<td>public</td>
</tr>
<tr>
<td><strong>Primary Production</strong></td>
<td></td>
</tr>
<tr>
<td>Spatial distribution chl $a$</td>
<td>Kim Hyde</td>
</tr>
<tr>
<td>CPR Dino/ Diatom</td>
<td>Jon Hare</td>
</tr>
<tr>
<td><strong>Secondary Production</strong></td>
<td></td>
</tr>
<tr>
<td>ZOOECO BV</td>
<td>Jon Hare</td>
</tr>
<tr>
<td>Copepod species</td>
<td>Jon Hare</td>
</tr>
<tr>
<td>Gelatinous zooplankton</td>
<td>Jon Hare</td>
</tr>
<tr>
<td>Other zooplankton</td>
<td>Jon Hare</td>
</tr>
</tbody>
</table>
Support for Ecosystem Status Report

Moving Forward

Use Scientists for Science

Use Computers for Routine Actions

The ESR Software App supports the routine “collection”, preparation, analysis and graphing of indicators in the ESR that also enhances Traceability, repeatability, explanation, verification, and validation
Data-to-indicator pipeline diagram

In the diagram below, arrows provide relationships, or ontology, between concepts. Broad arrows in the pipeline are labeled with steps defined below for data provenance. Dashed lines indicate overlap, or “fuzziness,” of concepts.

<table>
<thead>
<tr>
<th>Terms used in Ocean Health Index paper</th>
<th>metric from single data layer</th>
<th>metric from multiple data layers</th>
<th>dimension</th>
<th>goal score</th>
<th>“Ocean Health Index”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coupled natural human system</td>
<td>measured quantity that may be repeated in time and/or space</td>
<td>human-understandable quantity that may be repeated in time and/or space</td>
<td>combination of measured quantities</td>
<td>human-ununderstandable but can not be directly measured and requires time and/or space</td>
<td>dependent variable output of model using combination of measured quantities</td>
</tr>
</tbody>
</table>

NASA data levels 0,1
NASA data levels 2,3
NASA data level 4
Data provenance:
Adequate provenance record requires metadata from all the steps for trace-ability and repeat-ability

A: Acquisition metadata
B: Metadata for processing raw data into quantity that can be a metric (e.g., apply calibration)  
   ISO 191** standard
C: Metadata for combining source datasets into metric
   C1: Combination without application of model (e.g., ratio)
   C2: Combination with application of model with underlying assumptions
D: Metadata for producing an indicator from metric(s) (so far this step also includes combining metrics)
   use other metadata standard, e.g., SDMX
E {E1, E2}: Metadata for producing a composite indicator from multiple indicators
Support for Ecosystem Status Report

Moving Forward

Work with you to identify “source data” and work with prep, analysis and graphics code

Comments / Questions??

This Software App is scalable and elements could be used for other EcoAp Products
Choice for next iteration of the Use Case cycle:
- Continue development of C++ web app GUI?, or
- Develop Python desktop app GUI?

In both cases, the GUI will call to data and software library on server (iPython kernel), and high-level user of iPython notebook can perform tasks without the GUI

<table>
<thead>
<tr>
<th>C++ web app GUI</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No need to install app locally</td>
<td>- Less functionality in reproducing plots from ESR (will have to “feed” parameters to Python kernel)</td>
</tr>
<tr>
<td></td>
<td>- Quick-and-easy interactive visualization</td>
<td>- May need to hire C++ programmer during development</td>
</tr>
<tr>
<td></td>
<td>- Not as good with multiple users at same time</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Python desktop app GUI</th>
<th>Pros</th>
<th>Cons</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>- Same functionality as Python in reproducing plots from ESR</td>
<td>- Have to install app locally</td>
</tr>
<tr>
<td></td>
<td>- Massimo can develop fully</td>
<td>- Interactive visualization may not be as quick-and-easy as web app</td>
</tr>
<tr>
<td></td>
<td>- Better for multiple users at same time</td>
<td></td>
</tr>
</tbody>
</table>

- Better for multiple users at same time