COVE: a visual environment for ocean observatory design

Keith Grochow¹, Mark Stoermer², Deb Kelley³, John Delaney³ and Ed Lazowska¹

¹Department of Computer Science, University of Washington, Seattle, WA 98195, USA ²Center for Environmental Visualization, University of Washington, Seattle, WA 98195, USA ³School of Oceanography, University of Washington, Seattle, WA 98195, USA

E-mail: keithg@cs.washington.edu, mstorm@u.washington.edu, dskelley@u.washington.edu, jdelaney@u.washington.edu, edl@cs.washington.edu

Abstract. Physical, chemical, and biological ocean processes play a crucial role in determining Earth’s environment. Unfortunately, our knowledge of these processes is limited because oceanography is carried out today largely the way it was a century ago: as expeditionary science, going to sea in ships and measuring a relatively small number of parameters (e.g., temperature, salinity, and pressure) as time and budget allow. The NSF Ocean Observatories Initiative is a US$330 million project that will help transform oceanography from a data-poor to a data-rich science. A cornerstone of this project is the deep water Regional Scale Nodes (RSN) that will be installed off the coasts of Washington and Oregon. The RSN will include 1500 km of fiber optic cable providing power and bandwidth to the seafloor and throughout the water column. Thousands of sensors will be deployed to stream data and imagery to shore, where they will be available in real time for ocean scientists and the public at large. The design of the RSN is a complex undertaking, requiring a combination of many different interactive tools and areas of visualization: geographic visualization to see the available seafloor bathymetry, scientific visualization to examine existing geospatially located datasets, layout tools to place the sensors, and collaborative tools to communicate across the team during the design. COVE, the Common Observatory Visualization Environment, is a visualization environment designed to meet all these needs. COVE has been built by computer scientists working closely with the engineering and scientific teams who will build and use the RSN. This paper discusses the data and activities of cabled observatory design, the design of COVE, and results from its use across the team.

1. Introduction

The oceans have always been an important focus of scientific study and exploration. They cover over 70% of the Earth’s surface, moderate its climate by being a stabilizing force for the environment, and are a significant source of food for a large part of the planet. Because of the increasing impact of global warming, understanding ocean processes is increasingly central to predicting how our climate will evolve during the next century, and the impact on ocean and human health. To date, our ability to collect data about the oceans and simulate important interlinked processes has been extremely limited compared to the need.

A relatively new solution to this problem is ocean observatories that continually collect and analyze diverse data from the oceans in real time. Over the next few years, the National Science Foundation (NSF) Ocean Observatory Initiative (OOI) will create an ocean observatory of unprecedented scale. The Regional Scale Nodes (RSN) portion of the OOI, installed off the Washington and Oregon coasts, will support sensors from the ocean surface to deep in the seafloor, connected to cables delivering power and
bandwidth. The goal of this endeavor is to build a flexible platform allowing hundreds of scientists from various fields to conduct experiments, provide real-time sensor and data access through the Internet, and create a vast archive of data. The diverse range of users and unprecedented collection of data require a system that makes design and exploration of the observatory layout and experiments easy and intuitive.

Designing the observatory requires several tasks. Observed data for the sites must be collected and simulation models generated to understand the processes and possible issues at each experiment location. Different options must be explored and analyzed to determine optimal layouts for data collection within budgetary and technical constraints. The reasoning for alternative layouts must be presented to the science users to determine the effectiveness of various designs. These tasks must be carried out, not only to create the infrastructure, but regularly over the life of the system as science teams extend the observatory with new sensors.

Figure 1. COVE: Common Observatory Visualization Environment.

Visualization is particularly important to these tasks for several reasons. Sparse observed datasets require human expertise to fill in gaps and identify important trends. Oceanographic models require domain experts to assess their validity. At a proposed experimental site, scientists from diverse fields will be dealing with myriad new data types, planning 10–15 year experiments with hundreds of sensors in the context of existing seafloor bathymetry (seafloor terrain), sensor layouts, and environmental parameters (e.g., lava morphology, areas of hot spring venting, and gas hydrate deposition). Effective visual tools are crucial to allow human cognition to rapidly determine patterns, note anomalies, and determine the validity of data based on environmental context.
Although several tools provide solutions to some of these problems, none effectively solves them all. Georeferencing browsers such as Google Earth and Microsoft Live Earth (hereafter referred to as geobrowsers) provide a familiar physical metaphor for visualizing such large geographic layouts. Their multi-resolution interface affords immediate context for data, allows easy layering of datasets, and makes georeferencing data intuitive. But to date they do not support changing 3D bathymetry and deliver little visual analysis capability, limiting their usefulness for oceanographic investigation. They also have limited ability for direct manipulation of sophisticated cabled layouts and sensor deployment planning. Tools such as Fledermaus\textsuperscript{TM} or GeoZui3D allow importing and manipulation of bathymetry but do not have tools for rich interactive data exploration in scientific experiments or sensor layout.

Observatory designers must therefore currently move between multiple systems to create, analyze, and visualize their designs. This situation leads to inefficiencies in exploration and planning, makes sophisticated analysis pipelines seldom reusable, and leaves little opportunity to share processes or data. This scenario will be exacerbated when the RSN moves beyond the initial design phase and enables large-scale participation by the global science community. A system that integrates support for the above areas enhances planning of complex seafloor and water column experiments, science-engineering integration, and knowledge sharing by avoiding the need to use several different applications and interfaces.

COVE, the Common Observatory Visualization Environment, is a visualization system that provides a solution to these problems (figure 1) Based on investigation into the visualization needs for large-scale ocean observatory design, COVE was developed with scientists and engineers to meet those needs. Building from the geobrowser model, historical datasets for the sites are provided on layers to the user for inclusion in their current view. Behind these layers is a flexible web service-based architecture and an XML-based data handling and tagging scheme to make shared storage and access of geolocated historical data easy. New instrument layouts and extensions of core infrastructure can be easily applied with direct manipulation techniques and shared among the design team. Visualizations explaining these layouts can be shared for collaborative design discussions across the team or for presentation to diverse groups to explain decisions.

COVE has been used extensively by the team to create the core observatory design for the RSN. It has provided a common ground to investigate and present those designs to experts from across various fields of oceanography and earth science. Plans are being made to extend its capabilities for continuing work on the observatory in areas of operations and education and ultimately visualization of real-time datasets as they stream from the observatory.

2. Related work

Our general approach is closely related to the work by Wood et al for interactive visualization of a large spatio-temporal dataset [1], where Google Earth was used as a front end to allow users to quickly mash up geovisualizations. Geobrowsers such Google Earth [2], World Wind [3], and Virtual Earth [4] provide a very accessible interface to geographic data. Their focus on the earth globe paradigm and zooming interface have made them popular for viewing simple geospatial data. They facilitate labeling of locations, georeferencing maps and images, and sharing through XML-based documents. The Google Ocean project [5] by Magic Instinct Software provides an example of using geobrowsers for oceanographic data and shows the limitations of their use: bathymetry is limited to 2D images overlaid on the ocean surface, there are no data display or analysis tools, and there are no interactive layout capabilities.

Geobrowsers have been used to display Earth observatory data, as demonstrated by the EarthScope project [6]. They display seismometer locations with sensor detail available by clicking on a geolocated object. World Wind from NASA was used to provide a more extensive observatory solution through its plug-in interface [7]. Spyglass [8] is an observatory-specific system that supports viewing of instruments, their connections, and detail about specific nodes. It allows visualization features such as interpolation of real-time data values through a plugin interface. Geospatial context for the nodes is limited to a 2D map layer.
Two desktop tools for visualizing general oceanographic data currently available are the Interactive Data Viewer (IDV) [9] and GeoZUI3D [10]. IDV was built for the atmospheric community on top of VisAD. It supports sophisticated 3D model visualization (isosurfaces, contours, volumetric), terrain maps, and data animation but with no inherent support for oceanographic data. GeoZUI3D [11] (and its commercial version Fleidermaus) integrates multiple 3D bathymetry sets and geolocated 2D and 3D objects for oceanographic visualization and recently added the ability to play back time-varying 3D point data. Both these tools have sophisticated user interfaces that are often difficult for scientists to use and do not provide interactive layout tools.

3. COVE
To create a visualization system that would meet the unique needs of this user and data environment, we worked closely with the ocean observatory design team for several months to understand their requirements and develop possible visualization solutions. Our approach involved collecting artifacts, recording events, and carrying out interviews with the primary participants on the team. Current software and nonsoftware solutions were analyzed to determine possible improvements to the systems in use. Based on this investigation, prototypes were created to test hypotheses. We found that constant prototyping was an extremely useful tool to bridge the gap between various views of the best solution.

With a project of this scope, a team of diverse users was involved in the design decisions at various stages, which we grouped into three primary user categories:

- **Core Design Team**: a small group of scientists that creates a conceptual layout of the system to define infrastructure for the major science questions explored by the observatory
- **Engineering Team**: a group that integrates with the Core Design Team to generate technical analysis, itemized costing and implementation schedules
- **Principal Investigators**: individual scientists who extend the infrastructure by adding sensors to collect data for their specific experiments.

Driven by the design schedule for the RSN, our work to date has focused primarily on systems to aid the Core Design Team. The design process in place before COVE included use of multiple software applications and digital mapping tools that were not integrated. Proposed designs were created by the project team and captured using word-processing or spreadsheet documents and static visualizations that were manually created. These visualizations were used for community discussions and reviews. When necessary for outside communication, an onsite graphics group was employed to design high-resolution visuals.

Several limitations were identified in this process. The design solutions were hard to share among team members since they were often using different tools. Changes were time-consuming and expensive since there was no automated way to update costing or cabling. Comparison of different models required reviewing spreadsheets or paper visuals. It was difficult to view experimental sites from different angles or at different resolutions. Moreover, it was not possible to overlay key information on a new layout, such as bedrock characteristics or pre-existing cable route information, to easily determine design flaws.

Working with the science team, several requirements were identified for a more usable system to address these needs. The goal was an integrated architecture that would address all three of primary requirements areas:

- **Collecting and Registering Data**: All data needed to be in a common environment, rapidly accessible, and viewable in different ways so that it could easily be used across the team and visually checked for accuracy.
- **Creating Observatory Designs**: All infrastructure (nodes, cabling, connectors) and sensors, associated metadata, and costing needed to be integrated into the design so that technical limitations and cost-design benefits were readily visible.
• **Communicating and Collaborating**: Layouts and views of experimental/sensor sites needed to be easily sharable with mechanisms for comments by other users and presentation to diverse audiences.

To meet these requirements we created COVE, shown in figure 1. COVE is an immersive, direct manipulation 3D environment based on the geobrowser model for easy layering of georeferenced bathymetry, images, and datasets. It provides easy mechanisms to create, edit, and share sensor layouts with the team and science user community. And it allows histories of changes and decisions to be captured for later availability. Each of these areas is discussed in more detail below.

### 3.1. Architecture

The primary goals of our architecture were to provide (1) rich cross-platform, interactive 3D graphics, flexible internal design to quickly prototype new solutions and (2) the ability to use the web to transparently scale our data-handling needs.

We had hoped to use an existing geobrowser as the basis for our work on COVE, but we found that because of limitations in bathymetry and programming accessibility (Google Earth) or lack of cross-platform support at the time (World Wind), existing clients were not an option. Web browser-based solutions were also not viable since they did not provide the level of 3D interaction required. COVE was therefore implemented as a C++ open source project on top of the OpenGL graphics interface. A cross-platform interface was built using the Fast Light Tool Kit (FLTK). COVE is designed in a modular manner to allow porting to geobrowser platforms as it becomes feasible.

To provide flexibility, we use XML for both file formats and internal record handling. As a file format, this was an obvious choice based on existing geobrowser formats. For record handling, we expose the capability to read and write the XML text for a record at any time. This allows the state of a record to be viewed at any point as text output and edited as text in the UI without needing to continually add new dialogs. Another advantage is that undo/redo is easily handled by saving text versions of records at each modification. Further, users can add arbitrary tags to any record to handle unanticipated data needs without an update to the system.

### 3.2. Collecting and registering data

A first step in observatory design is to gather datasets providing information about the proposed infrastructure and experimental locations. These include variable resolution bathymetry datasets for underwater terrain, geological maps, images and video of the seafloor, observed and simulated datasets, and site features such as existing telecommunication cables, navigation hazards (e.g., shipwrecks) and legally protected zones.

A zoomable geobrowser interface allows the user to interact with datasets across the entire observatory while retaining context. It provides a viewable environment from the thousands of square miles covered by the RSN, down to a few meters around a specific sensor in an experiment. It also provides a simple layering metaphor for the data to help organize different data types. Figure 2 displays an example of different layered datasets in the same area of the RSN. The first image shows existing buoy locations and surface geology. In the second image, these datasets are turned off or made semi-transparent to examine recorded earthquakes more closely.

Bathymetry for the seafloor comes in a large variety of resolutions: from low resolution 100 km sampled sets based on satellite data, down to 1 meter samples. The terrain sets are commonly in flux as new versions become available, and certain terrain sets are shared only among a small group of scientists, so arbitrary file-based terrain handling is necessary. Our terrain engine uses a quad-tree based tiling system and a greedy algorithm to find the highest resolution point that can be interpolated from the currently loaded datasets. Where bathymetry for an area is undersampled, Catmull Rom splines provide a smoothing function for the surface to avoid faceting of the final terrain mesh. Maps and images can be overlaid on the terrain, as well as user-selected gradient sets with contours or binned color gradient values to highlight depth ranges.
Figure 2. COVE provides layering to quickly review relevant datasets. Here surface geology and buoy locations are easily replaced by earthquake fault maps and recorded seismic events.

COVE provides simple geolocated visualization of observed and simulated data over time. The goal is not to deliver the exhaustive capabilities of scientific visualization packages but rather to provide a set of interactive visual techniques most commonly used by the scientists. Data can be viewed as points, lines, vectors, planes, or isosurfaces using gradients to define the data color. The available visualization methods are expected to be expanded in the future as new scientific needs arise.

3.3. Creating observatory designs
The next step is to facilitate easy generation of sensor layout and cabling. COVE provides a direct manipulation drag-and-drop interface to add and position instruments. The instruments can be represented by icons, common 3D shapes, or instrument specific 3D models. Information summaries for an instrument can quickly be viewed in a popup window (make, model, data format, power consumption, etc.), or the user can be redirected to a detailed web page for the instrument. Based on metadata entered for the sensors, the system has the ability to detect intersensor interference and visualize the interference range.

Figure 3. An example of a detailed layout off the Hydrate Ridge site.

Figure 4. Here we have an annotation popup for communicating with a colleague about a site.

Default cabling is provided between new instruments and is automatically updated as instruments are moved. The cables properly follow the bathymetry, and problems such as extreme depth change can be detected and examined in a cabling window, presenting a cross-section view of the cable layout. Cable
handles can be arbitrarily added and positioned for maneuvering the cable around obstacles in the terrain or changes in the surface geology of the seafloor. For areas with insufficient bathymetry to completely determine cable positioning, seafloor surveys can be planned by directly dragging a survey field over the area in question.

A set of instruments and cabling can be removed or inserted using common cut, copy, and paste procedures, as well saving them under an instrument template. Templates allow easy insertion of complex instrument packages, such as mooring-based sensors or geodetic arrays. A subset of the layout can also be moved to another layer, making it easy to build up the observatory a section at a time. This also enables different designs to be viewed on top of each other to quickly determine differences. Session-based undo/redo is always available, making it safe and easy to explore different layout options.

Site metrics such as cost, power needs, and bandwidth on user-selected sections of the observatory, are automatically updated on the screen for the user during these editing sessions in a heads-up display. By having budgets readily visible and costs automatically updated with changes, layouts can be quickly refined to meet specific goals (figure 3). Moreover, detailed reports can be viewed in a table format or saved to a file for further analysis in tools such as spreadsheets.

3.4. Communicating and collaborating
Since the observatory is built by a team of scientists and engineers from many different disciplines and locations, sharing information across the team is crucial. COVE offers a set of tools to facilitate effective communication and collaboration. For simple file-based communication, users can always transfer their current work file to another user. Necessary datasets supporting the file layers are pulled from the data repository as allowed by access control. For communication outside the team, COVE allows output to high-resolution images (limited only by graphics memory), movies, and KML files that can be displayed in readily available programs such as Google Earth.

When scientists are satisfied with their changes locally, they can share the design to the data repository, where it is subject to simple check-in and check-out procedures to ensure integrity. In order to allow multuser editing of the observatory, it can easily be split into multiple layers with different users working on each one. To be kept apace of changes in the repository, users can register to receive messages in COVE (much like an RSS feed), should someone make a change to layouts or data that they care about. They can also register interest in a specific location and be informed of any new occurrences or posts.

New views can easily be created and saved by users to invoke a specific set of camera, layer, and visual settings. These can be used locally for examining work from various points of view or posted to the data repository for sharing with the team. Once a view is posted, another user can investigate and click on the view to activate it in COVE. Views can also be automated, like a slide show, to provide a guided tour through different aspects of the observatory. These views are available as thumbnails in the interface for easy selection and navigation through the observatory.

Users also can place annotations at any location or time in COVE. These annotations are represented visually by a familiar thumbtack icon and include the user, time, and urgency of the comment. Users can click on these to see what has been shared by their colleagues and then add comments if they wish. Because these annotations are stored in the repository with other team data, they are available on the web as well as through COVE and can be filtered and sorted to quickly find relevant information.

4. Results
The results for COVE were positive on many fronts and demonstrate the potential for this type of observatory design tool.

We observed many interactions that highlighted the value of integrating visualization tasks. Scientists would create new layouts and then use saved views to evaluate the results from various angles, particularly between 2D top-down and 3D perspective views. Maps of fault lines and surface geology were displayed to help with interactive cable layout, while using the heads-up display to ensure that
budgets were met. Since the terrain engine allowed collected bathymetry sets to be easily added, the team quickly built up a multigigabyte bathymetry repository for use when creating layouts. The scientists found the layering model for datasets intuitive and integration with Google Earth data formats a real benefit.

COVE was used to help create the primary infrastructure for the RSN in preparation for a recent NSF design review. It allowed different core cabling alternatives to be explored, and helped convince the team of the necessity for a significant change from a ring to a star based core configuration. It was also the key tool for creating visuals to explain the design at the review and to the other observatory teams. The review was deemed a big success for this team, and our users have been vocal in praising COVE as a key contributor to their achievements.

A very encouraging sign is the interest from user groups not part of our original study, who have asked that COVE be extended to handle their observatory visualization needs in areas such as daily operations and inventory tracking. We have also been approached by Earth sensor observatory teams to determine how to extend our solution to accommodate their needs.

Future plans include working aggressively to build in workflow and scripting capabilities. As noted above, these offer a scalable means to load and transform more datasets, while allowing COVE to be optimized for visual interaction. Workflow solutions are also well suited to real-time data flows inherent in the upcoming operations phase of ocean observatories. Another potential area of research is stronger integration with web browsers to determine the effectiveness of web-based collaborative solutions with COVE.

More Information about COVE
For more information about the COVE project, including available downloads of the most recent version, one can visit our website at http://www.cs.washington.edu/homes/keithg/oceans.html. More information about the Ocean Observatory Initiative and RSN is available at http://www.neptune.washington.edu/.

Acknowledgments
The authors thank Microsoft’s Technical Computing Initiative (particularly Roger Barga and Jim Gray), the University of Washington School of Oceanography, and the Monterey Bay Aquarium Research Institute for data and technical advice. This work was supported in part by grants from Microsoft Corporation and the National Science Foundation.

References
[1] Wood J, Dykes J, Slingsby A and Clarke K 2007 IEEE Transactions on Visualization and Computer Graphics 13 1176–1183 ISSN 1077-2626
[2] Google 2008 Google Earth http://earth.google.com
[3] NASA 2008 World Wind http://worldwind.arc.nasa.gov
[4] Microsoft 2008 Virtual Earth http://www.microsoft.com/virtualearth
[5] Magic Instinct Software 2008 Google Oceans http://www.justmagic.com/GM-GE.html
[6] Newman H B, Ellisman M H and Orcutt J A 2003 Commun. ACM 46 68–77 ISSN 0001-0782
[7] Coughlan J C and Hogan P 2006 AGU Fall Meeting Abstracts A811+
[8] Buschmann C, Pfisterer D, Fischer S, Fekete S P and Kröller A 2005 SIGBED Review 2 1–6 ISSN 1551-3688
[9] Unidata 2008 Integrated Data Viewer (IDV) http://www.unidata.ucar.edu/software/idv/
[10] Ware C, Plumlee M, Arsenault R, Mayer L and Smith S 2001 Proc. MTS/IEEE Conf. vol 3 pp 1960–1964
[11] Arsenault R, Ware C, Plumlee M, Martin S, Whitcomb L, Wiley D, Gross T and Bilgili A 2004 Proc. IEEE OCEANS Conf. vol 22 pp 743–747

8