The EventViewer: a tool for visualizing and exploring events extracted from Ocean Observing System Data

Kate Beard[1], Heather E. Deese,[3] Jake Emerson[1], Neal R. Pettigrew[2]

[1]Spatial Information Science and Engineering, University of Maine, Orono, ME 04469 USA

[2]School of Marine Sciences, University of Maine, Orono, ME 04469 USA

[3]Island Institute, Rockland ME, 04841 USA

***Abstract*-This paper presents capabilities of the EventViewer, a graphical user interface for visualizing and exploring patterns in events. The EventViewer supports queries on events stored in an events database and exploration of various temporal and spatial patterns in events. Interactions in the EventViewer interface allow users to select events from the database, assign spatial, temporal and thematic categories to graphic display elements called bands, stacks, and panels which causes events to be displayed according to their associations with the user selected categories. The spatial, temporal and thematic categories can be rearranged among the bands, stacks and panels thus changing the view of events and causing different patterns to become apparent. The EventViewer support exploration of periodic patterns, spatial and temporal trends, and event-event relationships. The EventViewer functionality is illustrated with oceanographic events extracted from the Gulf of Maine Ocean Observing system sensor data. Events are extracted from multiple time series variables collected at a number of locations and depths in the Gulf of Maine.**

I. Introduction

Oceanography has traditionally been limited by the expense and technical difficulty of collecting data at sea. Recent technological advances in autonomous sensors and platforms (including satellites, buoys, and floats) and government investment in permanent ocean observation systems are rapidly expanding the amount and quality of observational data. A key challenge in this new environment is to provide timely, meaningful interpretation of multiple data streams on a range of temporal and spatial scales to support oceanographic research, marine industry, and environmental management.

Arrays of sensors and sensor networks are contributing increasing volumes of multivariate spatio-temporal data and are likely to continue to expand, increase in size (extent of coverage), add new variables to the sensor suite, and increase in spatial and temporal sampling density, all of which will increase data volumes, dimensionality, and the challenges for data analysis and human assimilation. Different measurement protocols, wide variation in space time regimes, variations in media, formats, and data quality in observational sensor data streams challenge the ability to integrate and synthesize data from multiple sensors and platforms. Ocean observing systems that obtain data from a range of different sensors are clearly prone to these data integration and analysis challenges.

The ocean observing system for the Gulf of Maine [1], now operational for nine years http://gyre.umeoce.maine.edu/ or www.gomoos.org), has generated an unprecedented data record of very high temporal resolution data for several spatial locations. The observation system includes a moored buoy array with sensors measuring wind, waves, temperature, and visibility at the surface, and measurements of currents, temperature, salinity, ocean color, turbidity, light attenuation, and dissolved oxygen below the surface at multiple depths. Past and current moored buoy positions are shown in Fig. 1.

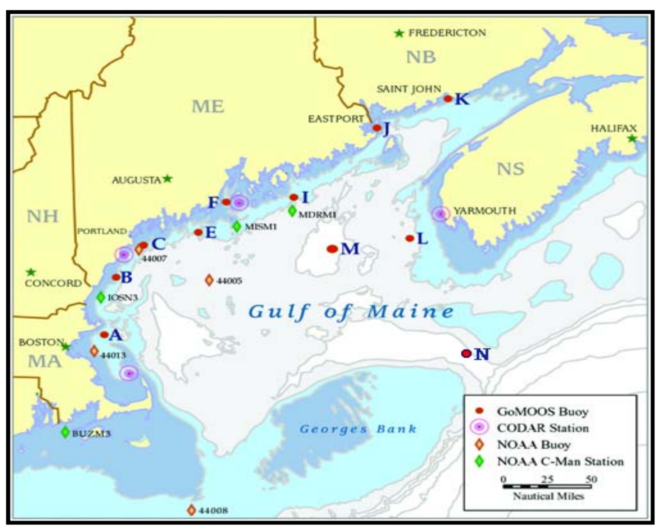


Figure 1. Locations of moored buoys in the Gulf of Maine.

Temporal sampling resolutions range from hourly to five and three minute intervals. Satellite data includes AVHRR sea surface temperature data collected at approximately 4-6 hour intervals at a spatial resolution of 1.1 km; NASA's SeaWIFs

978-1-4244-5222-4/10/$26.00 ©2010 IEEE

sensor measuring near-surface ocean phytoplankton biomass and composited as 8 day images; and wind data from NASA's SeaWinds instrument on board the QuikSCAT satellite measuring the speed and direction of near-surface winds at a spatial resolution of .25 degrees twice a day under all weather conditions. The system also includes data on sea surface currents from three CODAR (Coastal Ocean Dynamics Application Radar) stations.

While this represents a very rich data set, the analysis challenges are at least two fold. One challenge lies in the high dimensionality of the data. The Gulf of Maine ocean observing network has a structure similar to most sensor networks [2] which typically consist of multiple observation sites (sensor nodes) regularly or irregularly distributed in space. Each node or platform can have many sensors measuring different variables and producing a time series on each variable. Sensor network time series thus have three possible spatial dimensions, a temporal dimension, and numerous multivariate thematic dimensions.

The second challenge lies in the heterogeneity of sensor data. In the GoMOOS case, many of individual sensor streams, at, within, and above the surface, are incompatible with respect to media, formats, spatial and temporal resolutions, and measurement units, yet their synthesis is essential for building and advancing knowledge of ocean ecosystem dynamics.

The approach we have taken to the heterogeneous sensor data integration problem is to abstract sensor data streams to a common unit of analysis: an event data type. Work to date has focused on what we call primitive events, which represent statistically identifiable states or change states within individual observation data sequences. These primitive events can subsequently be synthesized into higher-level events for example oceanographic "events" such as wind-driven mixing, eddy formation, or current flow events. What one typically extracts as primitive events are key intervals of interest within a data sequence, such as periods of rapid change or exceptional sections of the data which might be a focus for deeper investigation. Transforming sensor data streams into primitive events has three useful outcomes: 1) it reduces dimensionality, 2) it isolates and characterizes particular changes of interest from an otherwise overwhelming environment of changing parameters so that users can focus on the relevant change, and 3) it normalizes across the heterogeneity of different sensor modalities by creating a new unified data representation.

Transformation of time series to event series reduces dimensionality but the resulting events have spatial dimensions (a horizontal position and sometimes height or depth corresponding to sensor platform positions), a temporal dimension, and a thematic dimension. Given their multiple dimensions, one would typically want to analyze them within this multidimensional context. To explore events from a sensor network, a user might wish to examine values from specific locations or for specific time periods. The set of locations at which events are assumed to occur are the sensor node locations or supersets of these. A user might want to see if the same type of events occurred at all sensor nodes, or investigate if events exhibit certain spatial trends (e.g. only occur at certain nodes or in spatial clusters). Additionally the user might want to investigate if certain types of events reoccur periodically, or if certain types of events show reoccurring patterns, e.g a type A event always occurs just prior to a type B event. Such explorations require an ability to examine event patterns in the dimensions of space, theme and time. The EventViewer has been designed to support interactive event visualization and exploration across these three dimensions [3]. As background, section II of this paper provides more detail on the structure of events. Section III then describes the organization and functionality of the EventViewer illustrating it capabilities using events extracted from the Gulf of Maine Ocean Observing network data.

II. The Event Model

People have developed labels to refer to common higher level events such as storms, droughts, traffic jams, wars, and plagues. These labels facilitate human ability to talk about such phenomena, store them, search for them, and otherwise interact with them in information systems. To model and work with entities in an information system assumes they have some handle, label, or documentation by which they can be effectively referred to. A primitive event in the context of this paper is a measurable change in the attribute values of a sensor data sequence. A number of methods exist to extract such entities from data sequences [4], [5], [6], but in general no commonly accepted vocabulary exists for consistently labeling such events. Some approaches attach labels for purposes of computational processing but without associating any semantics with the label [7]. To provide a common framework for describing primitive events, we developed a systematic approach for primitive event labeling referred to as an event model [8]. While not essential for viewing events in the EventViewer it establishes a common set of properties for primitive events by which they can be queried and viewed.

The event model is grounded in similar work on event detection methods [4], [5], [6], [9], [10], and identification protocols, which often involve identifying event start and end times based on observational values passing specific thresholds. Some of these methods similarly apply to the labeling and classification of higher level events (e.g. storms, hurricanes, tornados, earthquakes, volcanic events). For example, a hurricane event is identified once the one-minute average U.S. winds exceed seventy-four miles per hour and the hurricane is classified (in an adaptive fashion), based on the maximum wind strength relative to the 1-5 Saffir-Simpson scale. Similar event detection approaches from the oceanographic domain includes [9] who employ a time series segmentation technique to detect large phytoplankton biomass ‘events’ based on a ten year observational time series. They retain qualitative information about each event and are able to consider ‘recurrent events’ and anomalies in event patterns (e.g. due to climate variation), and [10] who illustrate a method to predict missing data and detect ‘exceptional’ events in a five hour observational time series.

The objective of the event model is to generate identifying properties for primitive events. There are at least three aspects to the primitive event identification problem: characterizing an event thematically, identifying its spatial extent, and identifying its temporal duration. The thematic identity and the temporal duration of primitive events are integrally linked. The spatial location and extent of all primitive events is simplified and assumed to be the location/footprint of the sensor node platform generating the data sequence. This location is assumed fixed. The thematic identity and the temporal duration of primitive events are integrally linked and generated from the event model. The event model includes four components: 1) observation data, 2) transforms, 3) extraction conditions, 4) and archiving conditions. The model is domain independent and applicable to many different types of events from individual sensor data sequences.

The event model generates descriptions for primitive events based on transformations applied to data sequences and extraction conditions that produce primitive events. Transformations apply to data sequences and can generate intermediate data sequences. Extraction condition functions take original or transformed data sequence as input and produce the primitive events as the output.

Transforms are part of standard approaches to processing data sequences and provide quality assurance, signal filtering, smoothing, discretization, differentiation, and other modifications to a data sequence. Observations from sensors are bound to contain spurious data, and these values can be smoothed with a running average, or trapped by a threshold and replaced using some stochastic method. Measured values are often influenced by phenomena acting on different time-scales. Transformations can filter out variance on time-scales that are not of interest. For oceanographic data, this may include removing long-term variability (e.g., seasonal or decadal cycles) or short-term variability associated with waves, tides, and diurnal cycles. Example transformations thus include smoothing, filtering out known signals, erroneous data elimination, conversion to rate of change (first derivative) or inflection (second derivative), among others. None, one, or many of these transforms may be applied, resulting in a modified data sequence. The event model includes transformations and transformation parameters as part of the primitive event identification properties. By adding the transformation history as part of primitive event descriptions, users are able to query for primitive events based on the transformation steps applied (e.g find all events that are result of a 10 hour filter) as well as display them by these properties.

The extraction conditions convert a data sequence to primitive event instances. An extraction condition function includes a pattern, order, minimum duration, and maximum break parameters. Examples of extraction condition patterns include a line (single threshold), band (two thresholds), model fit, or motif match. Order refers to the relative position of the observation data with respect to a pattern threshold. The order condition indicates for example that data values must be greater (e.g., above the threshold line) or less than (e.g., below a line) or inside a band. The minimum duration condition sets a temporal granularity threshold for events, e.g. events below the specified minimum are not included. This condition can filter noise that may not be considered significant. The maximum break condition serves to concatenate successive event instances separated by short duration interruptions. This condition parameter supports capture of events at coarser temporal granularities. An oceanographer for example may want to record salinity events on a monthly scale, and therefore might choose to ignore decreases in measured salinity lasting less than one day. Fig. 2 illustrates an single threshold extraction condition applied to a data sequence. Fig. 3. illustrates the same single threshold condition but with the addition of a maximum break parameter.

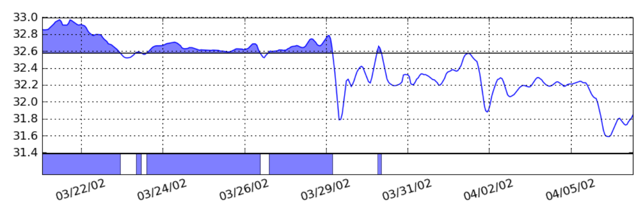


Figure. 2: Example event extraction condition where the pattern is a single threshold and the order condition is greater than the threshold. The bottom panel shows 5 extracted primitive events based on these conditions.

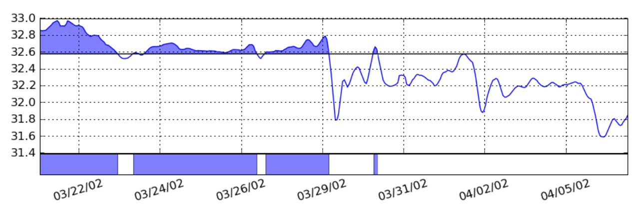


Figure. 3. Increasing the value of the maximum allowable break (“maxBreak”) between events concatenates two occurrences and decreases the number of extracted events to four.

The extraction condition parameters map each data point in a sequence onto the set {**true**, **false**}. A consecutive sequence of **true** values constitutes a primitive event instance which is assigned a unique event identifier. Primitive events obtain start and end times as a function of the applied event extraction process. The properties of a primitive event instance that are archived include a start time, and end time and a magnitude if appropriate. Primitive events extracted from the same data sequence with the same transformations and extraction conditions are considered events of the same class or an event type. The event type has properties that include the name of the measured variable of the originating data sequence, the set of transformation descriptors and the set of extraction condition descriptors. A user assigned label indicating event type may be included in the database.

Primitive event thematic descriptions can thus have a hierarchical structure that includes the variable type (e.g. salinity, temperature), event type (a function of transformation and extraction conditions), and event name. Primitive events can have additional metadata properties relevant to attributes of a sensor, or sensor network such as the sensor network custodian (eg. GoMOOS, Environment Canada).

A special type of event that is recorded for all time series is a missing data event. The properties of a missing data event include the location (sensor node), the data sequence variable, and the start and end of the missing data period. Missing data events thus have the same spatial, temporal and thematic attributes as their observed primitive event counterparts. Missing data events are displayed with the events from the same sensor data sequence to alert users that the absence of events in a pattern is due to missing data and not lack of event occurrence.

The event model was applied to the GoMOOS data for primitive event detection. The GoMOOS moored buoy array represents a typical sensor network structure in that it consists of a set of platform locations and at each location a set of variables are measured generating a set of n data sequences for each location. Therefore we have a set of locations L_s where there exist a set of primitive events $\{e_i^v\}$ of types $i=1,\ldots,n$ on variables $v = 1,\ldots,m$. As an example, for every buoy location where salinity is measured, there are sets of salinity events of various types depending on the transformations and extraction conditions applied. Selected thresholds utilized current oceanographic knowledge and operational definitions where they existed. A standard series of processing steps is outlined in Table 1.

TABLE 1

STANDARD APPLICATION OF THE EVENT MODEL FOR IDENTIFYING AND NAMING EVENTS IN GoMOOS OBSERVATIONAL OCEANOGRAPHIC DATA.

Observation Data	Hourly time series from oceanographic and meteorological sensors at sea surface and standard depths (Gulf of Maine Ocean Observing System buoy data)
Transform	Standard pre-processing: quality control, gap-filling Application of standard oceanographic low-pass filters. Application of transform specific to events of interest (see Table 2)
Extraction Conditions	Application of extraction condition specific to events of interest
Archive	Event characteristics of interest such as magnitude, variability, uncertainty of event parameters Include data sequence start time, end time, and missing data periods in event meta-data

TABLE 2

APPLICATION OF EVENT MODEL FOR IDENTIFYING FRESHENING AND SALTING EVENTS AT MULTIPLE TEMPORAL GRANULARITIES.

Observation Data	Transform	Extraction Conditions	Label
Salinity	Low-pass filter (33 hour); d/dt	Line=0 (less than)	Synoptic (2-10 day) freshening
Salinity	Low-pass filter (33 hour); low-pass filter (10 day); sub-sample daily; d/dt	Line=0 (less than)	Intra-seasonal freshening
Salinity	Low-pass filter (33 hour); low-pass filter (30 day); sub-sample 1 day; d/dt	Line=0 (less than)	Seasonal freshening

Table 2 shows an example application of the event model for extraction of salinity based events. The example show application of different transforms but use of the same extraction conditions across event types. The observation data is hourly salinity data from Buoy B at 1m depth. Basic pre-processing for qc and gap-filling is applied as a first transform step in all cases. The events extracted based on these transformations and extraction conditions are shown in Fig. 4.

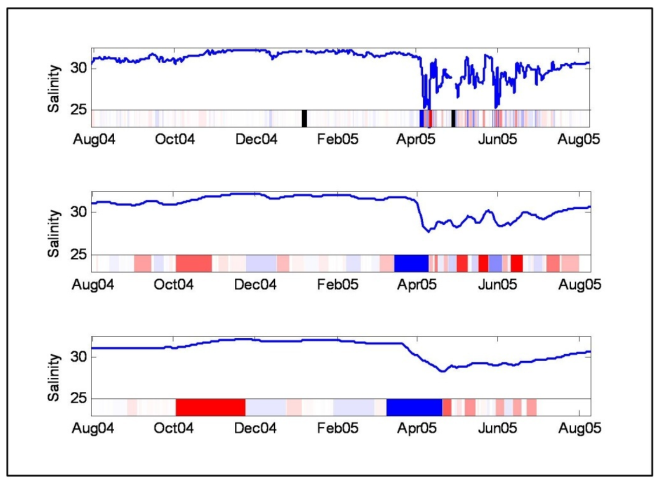


Figure 4: Freshening (blue) and salting (red) events at three different temporal granularities (accomplished by applying different low-pass filter and sub-sample processes in Transform step as indicated in Table 2 event model details.

The extracted events and their properties are stored in an events database implemented using Postgres and PostGIS. The EventViewer connects to this events database and can retrieve events by any of the described event properties.

III. THE EVENTVIEWER

Events as a new spatio-temporal data type require new tools for visualization and analysis. The EventViewer was designed to provide multi-dimensional space, theme, and time displays for events and support interactive visualization, exploration, and discovery of spatio-temporal event patterns. Standard maps portray measured spatial dimensions and a measured attribute (i.e implicitly every graphic mark on a map has a measured x, y position and a measured variable). Time series portray measured attributes and measured time. The space-time cube [11], [12] portrays measured time on the z axis and two measured spatial dimensions in x and y. Attempts to graphically represent measured space, theme, and time dimensions simultaneously present difficulties. The EventViewer uses a structured visualization framework [3] to address this problem by abstracting space, theme, and time dimensions to categories (labels) and associating these categories with sets of graphic elements in the interface called bands, stacks, and panels. The bands, stacks, and panels form nested graphic containers for displaying events symbolized by colored bars within a band (see Fig. 5). The width of the bar represents the duration of the event scaled to a time line associated with the band or stack. Different colors can indicate different types of events or properties of an event such as its intensity or magnitude.

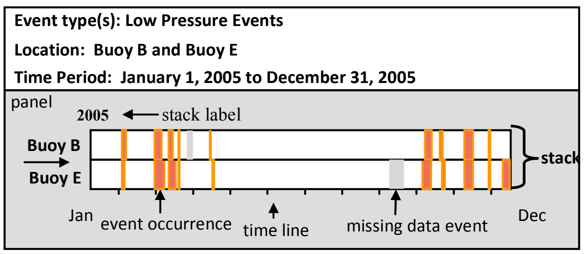


Figure 5. Example of event bands configured in a stack

The EventViewer framework supports the exploration of event patterns along the user's specified category assignments to bands, stacks, and panels. The user can assign space, theme, or time categories or other event metadata such as the transform or extraction conditions. The categories can form hierarchies or lattices that are functionally similar to the concept hierarchies in multidimensional databases [13], [14]. A temporal hierarchy logically consists of the common temporal units such as hours, days, weeks, months, and years. Location and thematic categories are more variable and application dependent. The location category lattice for the GoMOOS data is shown in Fig. 6.

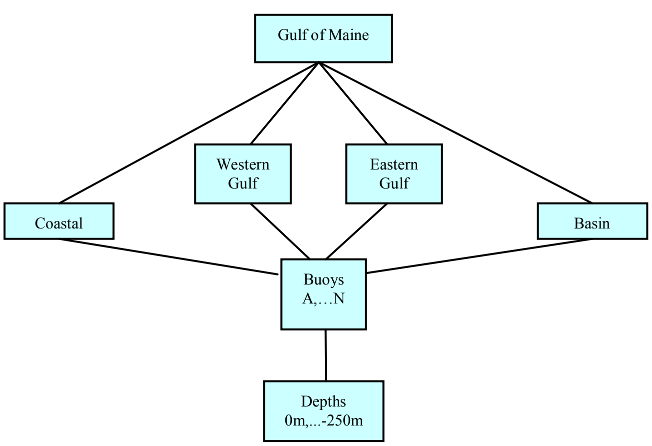
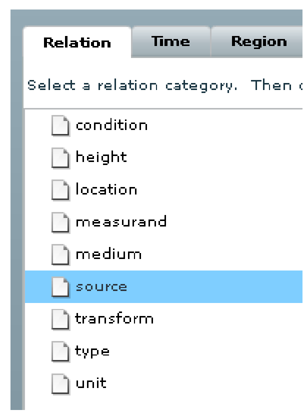


Figure 6. Lattice of location categories for the Gulf of Maine. The lattice defines how buoy locations are assigned to different gulf regions of interest.

To create a particular display, at least one type of event must always be selected. Given a selected event type, instances of the event type are mapped to bands, stacks, and panels according to how their properties map to the categories assigned to bands, stacks, and panels. Bands for example can be assigned to represent event locations (e.g. sensor node locations). The flexibility in the approach is that bands can just as easily represent a set of temporal categories (hours, days, weeks, months, seasons, years, lunar cycles, or even recurring events with time intervals such as El Nino), the thematic categories of events (barometric pressure drops and rises, precipitation events), or transformations and extraction conditions. The stacks and panels can be assigned categories in a similar manner and the structure is thus one that facilitates display of high-dimensional data. The three dimensions of space, theme, and time and the three graphic elements of bands, stacks, and panels allow for 27 possible assignments of dimensions to graphic elements providing a user with a wide range of options for exploring spatial, temporal, and thematic patterns in events.

The EventViewer opens with a query interface (see Fig. 7) in which the user first makes selections for types of events or event properties followed by selection of the categories to be assigned to the bands, stacks and panels. Types of events available for selection in the interface are generated directly from the population of events stored in the events database.



Figyre 7. Partial screen shot of query interface indicating selectable categories which correspond to relations in the database.

Once the user has made some initial selections on events, a time period of interest and categories, their selected events are displayed in the graph view according to the assignment of selected categories to graphic elements. Retrieved events are rendered as colored bars within the bands, stacks, and panels consistent with the user assignment of categories to the graphic elements.

The bands, stacks, and panels form small multiples [15], repeating graphic structures with common graphic elements. The repeating structure of the small multiples formed by the bands, stacks, and panels allows users to perform visual comparative tasks [16]. The EventViewer display shown in

Fig. 8 shows mixed events where years are the temporal category assigned to bands. Buoy locations are assigned to stacks and panels correspond to depth categories (top panel is 1-20m, and bottom panel is 20-50 meters). This arrangement allows the user, within a stack, to compare how event patterns vary by year. Across the stacks the user can compare how mixed event patterns vary across the three buoy locations (Buoys E, F and L). Across the panels the user can compare how the mixed event patterns compare with depth. For example it is quickly apparent that there are fewer mixed events at depth with each location.

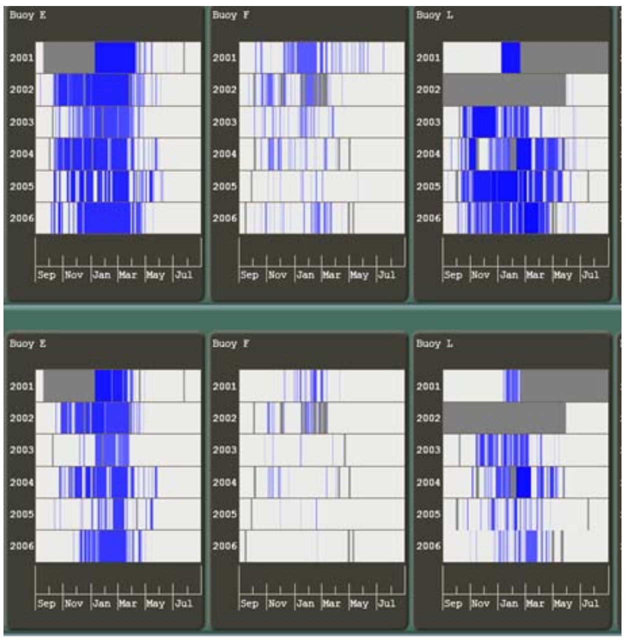


Figure 8. EventViewer graph illustrating mixed events at 3 buoy locations over 6 years and two depth difference categories (1-20m and 20-50 m)

The types of patterns that can be identified through various graphical arrangements and assignment of different categories to bands, stacks and panels include periodic patterns, spatial and temporal trends, and event-event temporal relationships (e.g. A always occurs just before B). Periodic patterns in events are revealed and explored by assigning different temporal categories (hours, days, months, seasons or years) to bands, stacks, and panels. For example diurnal patterns can be investigated by assigning days to bands and arranging the bands in stacks representing another temporal category such as weeks or months as shown in Fig. 9. Annual and other temporal periodicities are explored similarly by appropriate assignment of temporal categories to bands, stacks, and panels.

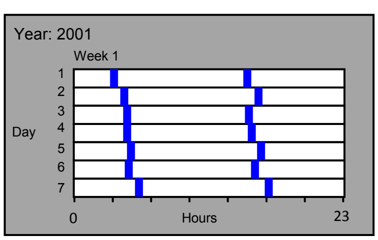


Figure 9. Example EventViewer graph illustrating band and stack temporal category assignments for examining diurnal periodicities.

Event-event relationships can be explored by assigning different event types to the bands as shown in Fig. 10. Such an arrangement allows users to investigate temporal relationships [17] among events such as one type of event commonly appearing just before, during, or after another event. Event-event relationships can be compared across different locations by assigning locations to the stacks or they can be compared by time by assigning temporal categories to the stacks.

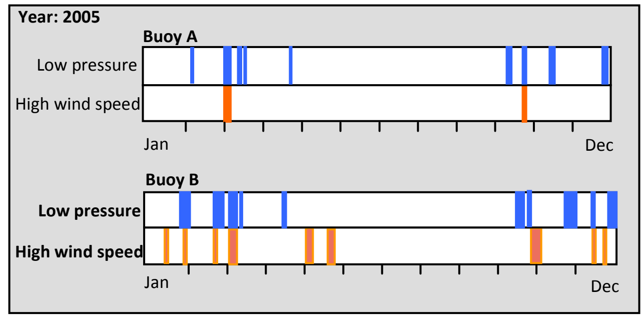


Figure 10. Example EventViewer graph illustrating event types assigned to bands to explore event-event relationships.

Spatial trends can be investigated by arranging spatial categories according to some spatial ordering. For example Fig. 11 includes increased flow events for major rivers emptying into the Gulf of Maine. The rivers are ordered south to north in the stack. In many case we can see that increased flow events occur concurrently across all seven rivers, but some spatial patterns are also evident. In June 2005 and June 2006, the strongest increased flows are seen at two neighboring rivers, the Androscoggin and the Kennebec Rivers. In December 2006, the southernmost rivers show sustained increased flow events while the northern rivers show only weak increased flow events.

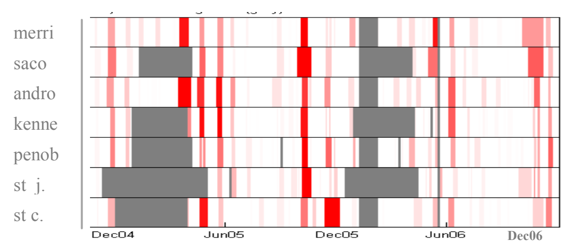


Figure 11. EventViewer graph illustrating spatial trends in increased river flow events. Color intensity indicates

The ability to quickly rearrange spatial, temporal, and thematic views on multi-dimensional data sets can be a powerful exploratory and pattern discovery tool for the user and new views may lead to specific patterns of interest that the user may want to explore further. The EventViewer allows rapid view changes through direct manipulation by dragging and interchanging the category labels among bands, stacks, or panels. For example Fig. 12 shows an arrangement with years as the temporal category assigned to bands and buoy locations assigned to stacks. In this view, a comparison of mixed events by year across locations is facilitated. Swapping the band and stack labels quickly changes the view so that a user can compare the mixed events at locations year to year more effectively.

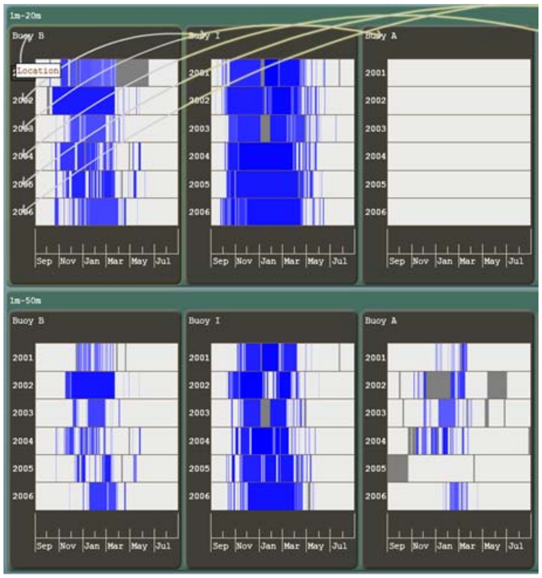


Figure 12a. The EventViewer graph shows mixed events temporal categories assigned to bands and locations assigned to stacks. This view facilitates comparison of mixed events at a location over a set of years. Swapping the band and stack categories results in a new view as shown in Fig 12. b. Mixed events are defined by a threshold (.125) on density difference with depth.

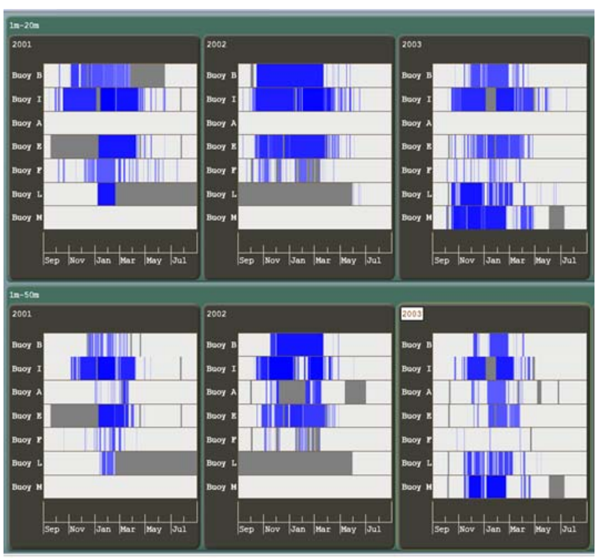


Fig. 12b This EventViewer graph show the swapped view from Fig 12 a. This view facilitates comparison of mixed events across locations over a set of years. Mixed events are defined by a threshold (.125) on density difference with depth.

The EventViewer provides a new tool for visualizing and exploring event data. It presumes the existence of event data. Events in this context are extracts from data sequences defined by a set of transformations and extraction conditions. However, the general requirements for an event data type are that it have a unique identifier, a thematic name , label or type, and a spatial location. This paper focuses on events extracted from the Gulf of Maine Ocean Observing System data sequences. Other types of events such as stock performance events from financial time series or traffic events from transportation monitoring systems.

The EventViewer loses some specificity in that it is structured around categories or qualitative values of dimensions rather than the measured values of dimensions. However, it is the organization and display of events by spatial, temporal and thematic categories as well as other metadata categories that provides the powerful exploratory basis of the EventViewer and the flexible framework for visualization and event pattern exploration. The EventViewer allows the exploration of events within a multidimensional space of categories of interest to the user and pertinent to the events (there must be a mapping from the event properties to the categories). If one is exploring traffic events, the location categories of interest might be street intersections, street segments, or traffic zones. In the oceanographic setting presented here, the user can specify location categories as any partitions of the Gulf of Maine region deemed of interest (e.g. basins versus coast or east versus west). Similarly users can specify temporal and thematic categories of interest. By modifying the spatial, temporal and thematic granularity of these categories coarser or finer views on events can be obtained. Finally direct interactions in the user interface make it easy to select events, select categories and quickly create and rearrange views.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 0429644 through the Science and Engineering Informatics and Information Integration Program (SEIII). Mike Scott and students in the ASAP program at the University of Maine are gratefully acknowledged for their work on the design and development of the EventViewer interface.

REFERENCES

[1] N.R. Pettigrew and C. Poesler, "Implementing the Gulf of Maine Ocean Observing System (GoMOOS)," Proceedings Oceans 2005-Europe, (Brest, France 2005). IEEE Press: Chicago, vol. 2, pp. 1362-1369, 2005.

[2] C-Y. Chong, and S. Kumar, "Sensor Networks: Evolution, Opportunities and Challenges," *Proceedings IEEE* Vol. 91.8, pp. 1247-1256, 2003.

[3] K. Beard, H. Deese, and N. R. Pettigrew. "A framework for visualization and exploration of events," *Information Visualization*, Vol. 7. pp. 133-151, 2008.

[4] V. Guralnik and J. Srivastava, “Event detection from time series data,” *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Diego, CA. pp. 33-42, 1999.

[5] Last. M. Y. Klein, and A. Kandel, 2001. Knowledge Discovery in Time Series Databases, *IEEE Transactions on Systems, Man and Cybernetics* Part B, 2.

[6] F. Höppner,, Time Series Abstraction Methods -- A Survey. In *Proceedings GI Jahrestagung Informatik, Workshop on Knowledge Discovery in Databases*, LNI, Dortmund, Germany, pp. 777-786, 2002.

[7] J. Lin, E. Keogh, and S. Lonardi, “Experiencing SAX: a Novel Symbolic Representation of Time Series,” *Data Mining and Knowledge Discovery*.Vol.15 no.2. pp. 107-144, October 2007.

[8] J. Emerson, H. Deese, and K. Beard, Identification of Primitive Events from Sensor Data Sequences, (in review).

[9] B. Beliaef, P. Gros, C. Belin, B. Raffin, I. Gailhard, and J-P Durbec, “Phytoplankton Events in French Coastal Waters during 1987-1997,” *Oceanologica Acta* vol. 24(5). pp. 425-433, 2001.

[10] F. Ibanez and A. Conversi, “Prediction of missing values and detection of exceptional events in chronological planktonic series: a single algorithm,” *Ecological Modeling* vol. 154. pp. 9-23, 2002.

[11] T. Hagerstrand, “What About People in Regional Science,” *Papers of the Regional Science Association.* Vol. 14. pp. 7-21, 1970.

[12] M-J. Kraak, “The Space-time cube Revisited from a Geovisualization Perspective,” *Proceedings, International Cartographic Conference*, Durban, South Africa, pp. 1988-1996, August 2003.

[13] S. Chauduri and U. Dayal, “An overview of data warehousing and OLAP technology,” *SIGMOD Record*, vol. 26. pp. 65-74, 1997.

[14] S. Shekhar, C. Lu, X. Tan, S. Chawla, and R. Vatsavai, “Map cube: a visualization tool for spatial data warehouses,” In: Miller H, Han J. (Eds), *Geographic Data Mining and Knowledge Discovery*, Taylor & Francis: London, pp. 74-109, 2001.

[15] E. Tufte, *The Visual Display of Quantitative Information* Chesire, CT: Graphics Press. 1983.

[16] J. Bertin, *Semiology of Graphics*. The University of Wisconsin Press, Translated by William J. Berg. (1972). 1983.

[17] J. Allen. “Maintaining knowledge about temporal intervals,” *Communications of the ACM*. Vol. 26. pp. 834, 843, 1983.