INDICATOR: An Open-Source Cyberenvironment for Biosurveillance

Wendy A. Edwards, M.S
Health Sciences Group, National Center for Supercomputing Applications, University of Illinois
wedwards@ncsa.uiuc.edu

Awais Vaid, MBBS, MPH
Champaign-Urbana Public Health District
avaid@c-uphd.org

Ian S. Brooks, Ph.D.
Health Sciences Group, National Center for Supercomputing Applications,
University of Illinois
ian@ncsa.uiuc.edu

ABSTRACT

In this paper, we discuss the architecture and implementation of INDICATOR, a free open source cyberenvironment for disease surveillance. Biosurveillance entails numerous tasks, including data acquisition and preparation, analysis, and reporting. These tasks can be modeled and executed as a workflow. Workflows encapsulate data, tools, and metadata.

Cyberenvironments provide integrated, user-friendly sets of tools and services to marshal resources and help researchers analyze, visualize, and model their data. INDICATOR uses an Eclipse-based cyberenvironment that supports interactive workflow creation, connection to data and event streams, provenance tracking, and reuse of workflows and fragments to acquire, analyze, and visualize public health data.

Keywords

Cyberenvironment, biosurveillance, open source, Java, public health.

INTRODUCTION

Healthcare professionals responding to a disaster or potential outbreak need rapid access to information to support a coordinated response. They need to know the size, spread, and location of any outbreak, and they may also require access to models to determine a containment strategy.

There are currently numerous software tools for outbreak detection, but these tools generally support a single type of data, and offer limited detection algorithms and little or no support for planning responses to outbreaks.

INDICATOR uses Cyberintegrator (Marini et al. 2006,), an Eclipse-based cyberenvironment (Myers and Dunning 2006), to manage biosurveillance-related workflows. A cyberenvironment is an integrated set of tools and services tailored to a specific discipline, Cyberenvironments offer data stores, computational capabilities, analysis and visualization, and access to shared instruments and sensor networks. Cyberintegrator supports interactive workflow creation, connection to data and event streams, provenance tracking, and reuse of workflows and fragments.

Provenance tracking refers to the problem of keeping track of where data comes from and the results of the activities in the workflow. For example, if we get a stream of emergency room department data and run a WSARE analysis, we need to maintain information about where the data came from, which algorithms were used, and where the results of the analysis are stored. Cyberintegrator uses Tupelo (Futrelle et al. 2009), software that stores provenance metadata in Resource Description Framework (RDF) format. Tupelo supports a context object, which represents a source or destination of information and acts as a broker between applications and this information, providing many different ways of storing, querying and managing data.

Reviewing Statement: This paper represents work in progress, an issue for discussion, a case study, best practice or other matters of interest and has been reviewed for clarity, relevance and significance.

METHODS

INDICATOR currently receives data related to patient advisory nurse (PAN) calls and emergency department visits from a local hospital, and school absences from Champaign County via email, which it parses, stores, and analyzes. It uses WSARE ("What's Strange About Recent Events") (Wong at al 2003) to analyze the PAN and ED data for anomalies. We have also used surveillance algorithms developed in R (Höhle 2007), an open source statistical package, to analyze PAN data related to common complaint codes and generate alarms. Currently, we are developing an interface that will allow users to use the R surveillance algorithms in Cyberintegrator.

Because Cyberintegrator supports both native executables and Java code, it is also be possible to run biosurveillance software released in executable format (not open source), such as SatScan (Kulldorff 1997). We have successfully run SatScan with Cyberintegrator and may do so again in the future if we receive appropriate data.

INDICATOR has a Liferay-based (Liferay 2008) web interface with portlets that display data, custom graphs, and analysis results related to PAN data. More recently, we developed an AJAX-based web interface to support data entry, management, and visualization of the school absence data. Because the amount of school absence data is limited at this point, it currently allows the user to specify a baseline time period for a particular school and generates an alert when the number of absences is more than three standard deviations greater than the mean for the baseline period.

Below is a diagram of INDICATOR's architecture (Figure 1). The workflow incorporates the data, metadata, and executable(s), and the system stores the metadata, data, and analysis results. This information is displayed in INDICATOR's web interface and is used to generate alerts.

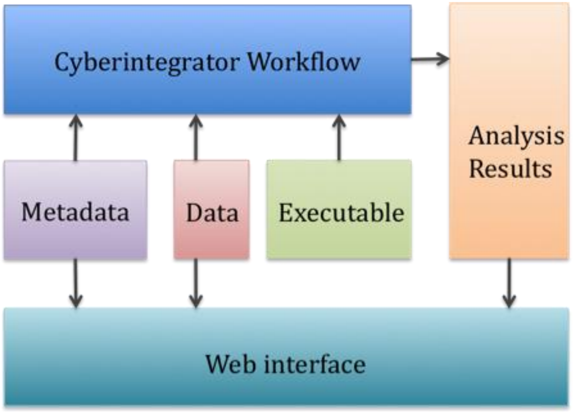

Figure 1 Diagram of INDICATOR architecture

RESULTS

INDICATOR has been online for over a year with patient advisory nurse data and has generated alerts with the WSARE algorithm that have corresponded to local public health events. We have more recently added emergency room and school absence data, and the WSARE algorithm has also successfully flagged anomalies in data related to emergency room diagnoses.

In the Fall of 2009, there was significant flu activity in Champaign County. There were corresponding elevations in school absences (Figure 2), flu-related PAN calls (Figure 3), and flu-related ED diagnoses (Figure 4). As the graphs show, the highest levels of activity for all three occurred in mid-October.

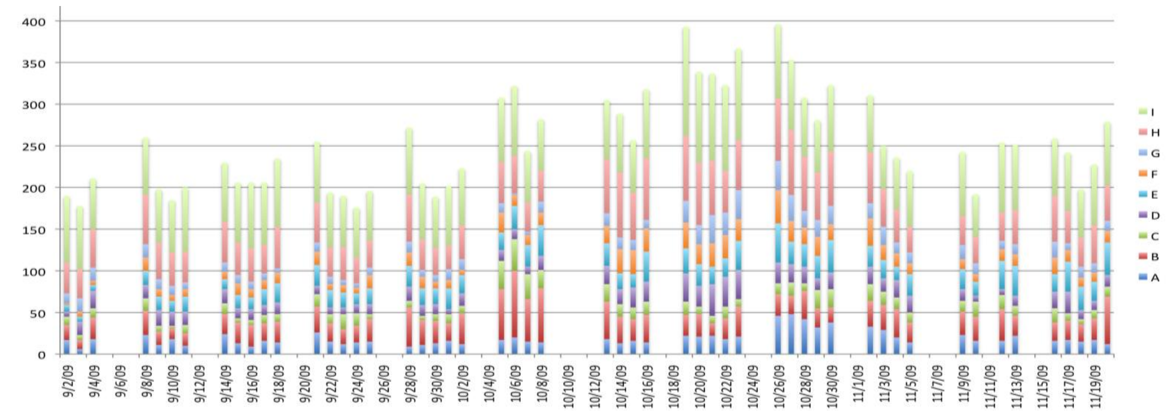

Figure 2 Urbana School Absences 9/2/2009 - 11/19/2009

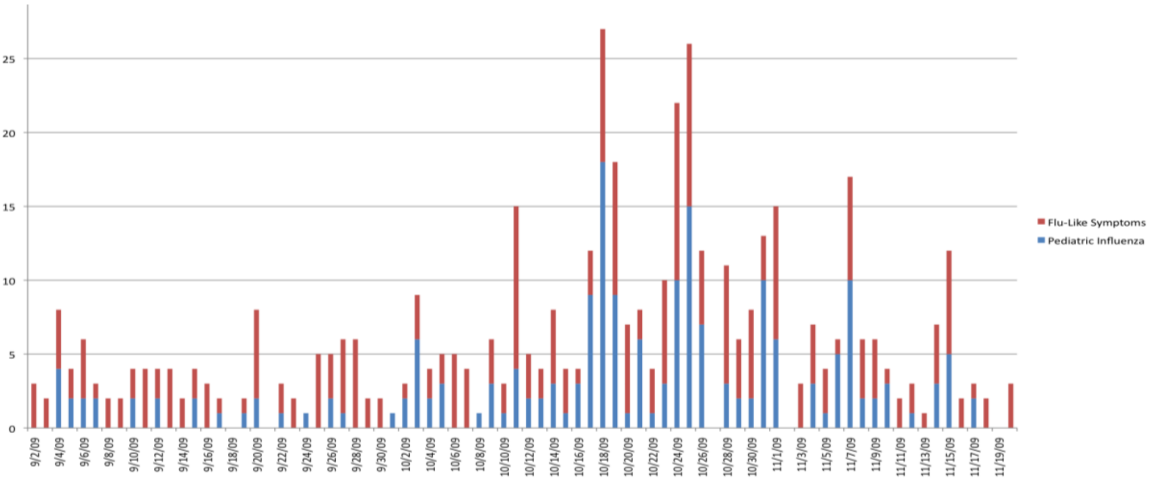

Figure 3 Flu-Related PAN Calls 9/2/2009 - 11/19/2009

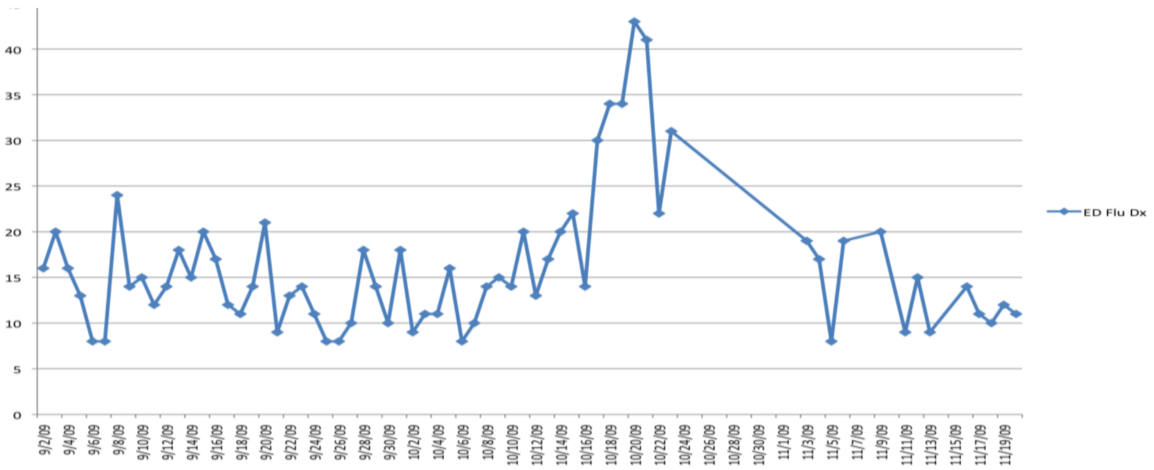

Figure 4 Flu-Related ED Calls 9/2/2009 - 11/19/2009

We have also used surveillance packages implemented in R to analyze the PAN data. These algorithms were more sensitive than WSARE, but generated reasonable alarms that corresponded closely to the data.

FUTURE WORK

There are numerous surveillance algorithms available, and it is not always clear to a health professional which ones are most appropriate for a particular set of data. For example, many require at least a year of historical data. Some deal better than others with missing data. We plan to add an interface to help users select an analysis algorithm that will work with their data.

We noticed that the ED data included a large vocabulary that makes WSARE analysis computationally expensive. We ran it through the National Institute of Health (NIH) Unified Medical Language System Knowledge Source Server (UMLSKS) metathesaurus (NIH 2009) and were able to automatically map almost all of the diagnosis values in our ED data. UMLSKR also supports hierarchies of terms; for example, "influenza" would fit into the MESH category of "Respiratory Tract Diseases." We would like to include support for automatic term mapping, This could be particularly useful when we are analyzing multiple streams of data and want to know whether terms from different sources have identical or similar meanings.

We are also interested in adding data mining functionality to INDICATOR to allow users to explore relationships between sets of data. For example, weather information may be correlated with the levels of certain complaints or diagnoses. PAN calls and ED diagnoses may be correlated with school absenteeism during an outbreak.

Finally, we plan to add modeling capabilities to INDICATOR and leverage NCSA's supercomputers to perform this computationally expensive task.

CONCLUSIONS

INDICATOR has successfully monitored streams of local public health data and generated alerts when anomalies were detected. The user-defined workflows offer unique potential, allowing users to define their own data sources and methods of analysis, specify reporting options, and view output online. Rather than competing with existing systems and analysis algorithms, INDICATOR provides a cyberenvironment to support them.

ACKNOWLEDGMENTS

We would like to thank Jim Myers, Joe Futrelle, Rob Kooper, Luigi Marini, Penny Meeker, Pam Manselle, William Schuh, Suzanne Stratton, Eugene Young, and Gina Keller

REFERENCES

1. Futrelle, J., Gaynor, J., Plutchak, J., Myers,, J. McGrath, R., Bajcsy, P., Kastner, J. et al. 2009. Semantic middleware for e-science knowledge spaces. Paper presented at MGC '09: Proceedings of the 7th International Workshop on Middleware for Grids, Clouds and e-Science, Urbana Champaign, Illinois.
2. Höhle, M. 2007. : An R package for the monitoring of infectious diseases. *Computational Statistic*s 22, (4) (12/01): 571-82.
3. Kulldorff, M. 1997. A spatial scan statistic. *Communications in Statistics - Theory and Method*s 26, (6): 1481.
4. Liferay - enterprise open source portal. 2008 [cited 7/9 2008]. Available from http://www.liferay.com.
5. Marini, L., Minsker, B., Kooper, R., Myers, J., and Bajcsy, P. 2006. CyberIntegrator: A highly interactive problem solving environment to support environmental observatories. *AGU Fall Meeting Abstract*s (dec): B908.
6. Myers, J. and Dunning, T. 2006. Cyberenvironments and cyberinfrastructure: Powering cyber-research in the 21st century. *Proceedings of the Foundations of Molecular Modeling and Simulation*, Blaine, WA.
7. NIH. UMLS Knowledge Source Server. Bethesda, MD, 20092010]. Available from http://umlsks.nlm.nih.gov/KSClassic/servlet/Turbine/template/admin%2Cuser%2CFactSheet.vm.
8. Wong, A., Cooper, G., Wagner, M.. 2003. What's strange about recent events. *Journal of Urban Healt*h 80, (June): 66-75.