OpenESSENCE: An Open-Source, Self-Contained Disease Surveillance Software Application for Global Use

Timothy C. Campbell, Charles J. Hodanics, Zarna S. Mistry, Adjoa M. Poku, Gabriel N. Gorelick-Feldman, Colin J. Taylor, Richard A. Wojcik, Jacqueline S. Coberly, Aaron T. Katz, Steven M. Babin, and Sheri H. Lewis

Electronic biosurveillance systems can improve the timeliness of public health data collection, aid in the early detection of disease outbreaks, and enhance situational awareness. As part of the Suite for Automated Global Electronic bioSurveillance (SAGES) program, the Johns Hopkins University Applied Physics Laboratory (APL) developed an open-source software tool called OpenESSENCE. OpenESSENCE provides "out-of-the-box" web-based data entry, analysis, and reporting that may significantly improve global disease surveillance, including surveillance in a wide range of resource-limited settings. Local health clinics have recognized that this new technology may help their countries comply with the World Health Organization revised International Health Regulations (IHR 2005) and prevent or mitigate disease outbreaks. This article briefly reviews OpenESSENCE and describes updates made during the last 2 years.

INTRODUCTION

Emerging public health threats can originate from almost anywhere in the world and often spread rapidly through populations, especially those lacking prior exposure to the disease in question.[1] Early detection of these disease outbreaks is considered so important to global health that the World Health Organization issued revised International Health Regulations (IHR 2005) to enhance global cooperation and protection from emerging health threats.[2] Recent advances in technology are helping member nations comply with IHR 2005. In 2010, Ashar et al.[3] reviewed information and communications technologies developed for electronic health data capture and assessed their use in resource-limited settings. To complement and enhance traditional public health surveillance, new syndromic surveillance systems were developed. These systems use electronic, nontraditional prediagnostic health information as early indicators of disease incidence to detect potential disease outbreaks in populations. More than a decade ago, the Johns Hopkins University Applied Physics Laboratory (APL) began developing a syndromic surveillance system called the Electronic Surveillance System for the Early Notification of Community-based Epidemics (ESSENCE).[4] ESSENCE is a Java-based web application used to monitor the health of populations, detect disease outbreaks early, and help prevent the spread of disease. "Enterprise" ESSENCE is the version of ESSENCE used by local and regional public health departments in the United States and by the U.S. Departments of Defense and Veterans Affairs.[5] It allows public health professionals to collect and analyze many types of nontraditional, prediagnostic data and to use multiple anomaly detection algorithms to flag unusually high counts of disease indicators or issue alerts. ESSENCE enables users to view, parse, plot, and map results and to share selected information with other users. Enterprise ESSENCE relies on electronic health data feeds and is best suited for areas with stable Internet access.

As part of the APL Suite for Automated Global Electronic bioSurveillance (SAGES) program,[6] the Global Emerging Infections Surveillance and Response System, a division of the U.S. Armed Forces Health Surveillance Center (AFHSC-GEIS), funded the development of a new version of ESSENCE, called OpenESSENCE (OE), for global use including use in resource-limited settings. As the name implies, OE is an open-source application that includes key features of Enterprise ESSENCE but can be used either via the Internet or as a standalone system.[6,7] Whereas Enterprise ESSENCE uses proprietary software, OE uses open-source software. Doing so provides several advantages in managing health data and performing medical surveillance in a variety of global communities. OE's open-source design makes it easier for global users to maintain and sustain their systems by providing better quality assurance, low cost for acquisition and maintenance, and extensive user input on requirements, usage, and adaptability.[8] In 2012, Campbell et al.[9] described the early development of OE and how it was being adapted globally to provide early warning and international awareness of disease outbreaks. This article will briefly review the basic features of OE and describe OE developments that have occurred since the Campbell et al. article[9] was written.

OE OVERVIEW AND UPDATES

Beginning early in the development of OE, APL sought input from a variety of public health departments in resource-limited countries and found that there was consensus on the desire for a new kind of biosurveillance system: an open-source system that would place a minimum burden on data providers, allow the user to tailor the system to their needs, and provide sustainability by allowing the user to control their data and easily maintain their system.

OE is designed to be customizable, dynamic, and flexible, allowing for extension and reconfiguration, to reduce the cost of development and maintenance. To further this goal, OE is deployable using only open-source software that includes industry-standard technologies (e.g., Java EE, Apache Tomcat, Spring, MySQL, PostgreSQL, PostGIS, GeoServer, GeoExt, and OpenLayers). OE uses the Spring (http://projects.spring.io/spring-security/)[10] framework and Groovy (http://groovy.codehaus.org)[11] Java language extensions to create a modular, component-based application design, which allows for improved testability of components and isolation of problems. Repurposing and reusing the application is easier and more likely with a modular design that mitigates the need to rebuild the application to incorporate adjustments or enhancements. To support confidentiality and integrity of data, OE provides user-configurable access control. Because it is intended for global use, OE features extensive localization support that allows for the adaptation of internationalized software for a specific region or language. OE includes a plug-in application programming interface that allows user access to different detection algorithms. Results are provided in a format similar to those in Enterprise ESSENCE and include graphs, charts, and tabular data on case reports, as well as geographic maps of individual illness reports.[4,5]

Initial User Interface and Data Input

Health indicator data are those data that reflect determinants of health and health outcomes and may come in a wide variety of formats that include numeric and textual data.[12] OE is designed to accept input of a variety of health indicator data.[13] OE contains a built-in data entry module so that data can be entered directly into a server running OE or via the web by multiple, geographically distributed users. The software is configurable to accept data from different databases and data sources and can accept combinations of different data sources (e.g., joins of multiple tables). At least two

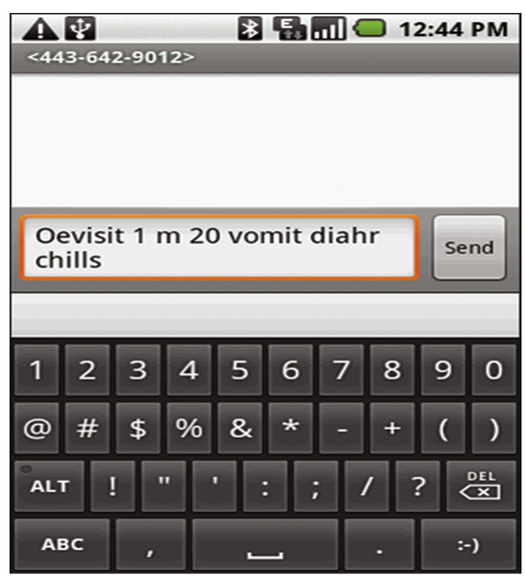


Figure 1. Example of simple structured SMS message for entering data in OE.

methods of data input are provided: (*i*) a cellular telephone Short Message Service (SMS) interface; and (*ii*) a web interface for those with Internet access. Both can be configured to accept data on the basis of individual collection needs.

In most countries, even those with limited resources, cellular telephone use is pervasive. Therefore, cellular services with the ability to transmit text messages using SMS can be used as a way of collecting health data and entering those data directly into an electronic surveillance system.[3,14] APL-developed SMS systems are currently used for OE data entry in multiple countries including Cambodia, Nicaragua, Cameroon, Kenya, and Uganda. Patient-level data, such as age, sex, presenting signs and symptoms, date of symptom onset, and local clinic identifier, are sent via SMS from local health clinics to a "receiver" Android smartphone. This receiver phone is connected to a computer that may be located at the central public health office, and the information is transferred to the computer using standard extract, transform, and load synchronization processes. Users can manually construct SMS messages to submit the data (Fig. 1), but this technique is tedious and error-prone. Additionally, the SMS standard[15] limits messages to a maximum of 160 characters. An alternative to manual SMS messaging is the form-based data entry application, SAGES mCollect, for Android smartphones. Using mCollect, which is powered by Open Data Kit Collect (http://opendatakit.org/use/collect), data are entered into a graphical, predefined form using a sequence of input prompts, which can include text, numbers, geo-locations, multimedia, and barcodes (Fig. 2). A multisegmenting feature developed by APL is used to overcome the SMS 160-character limitation. This feature divides the text message into 160-character segments for SMS transmission and also provides for message reassembly upon receipt by the receiver phone. A second form submission

Figure 2. Example of the mCollect form-based data entry interface for OE.

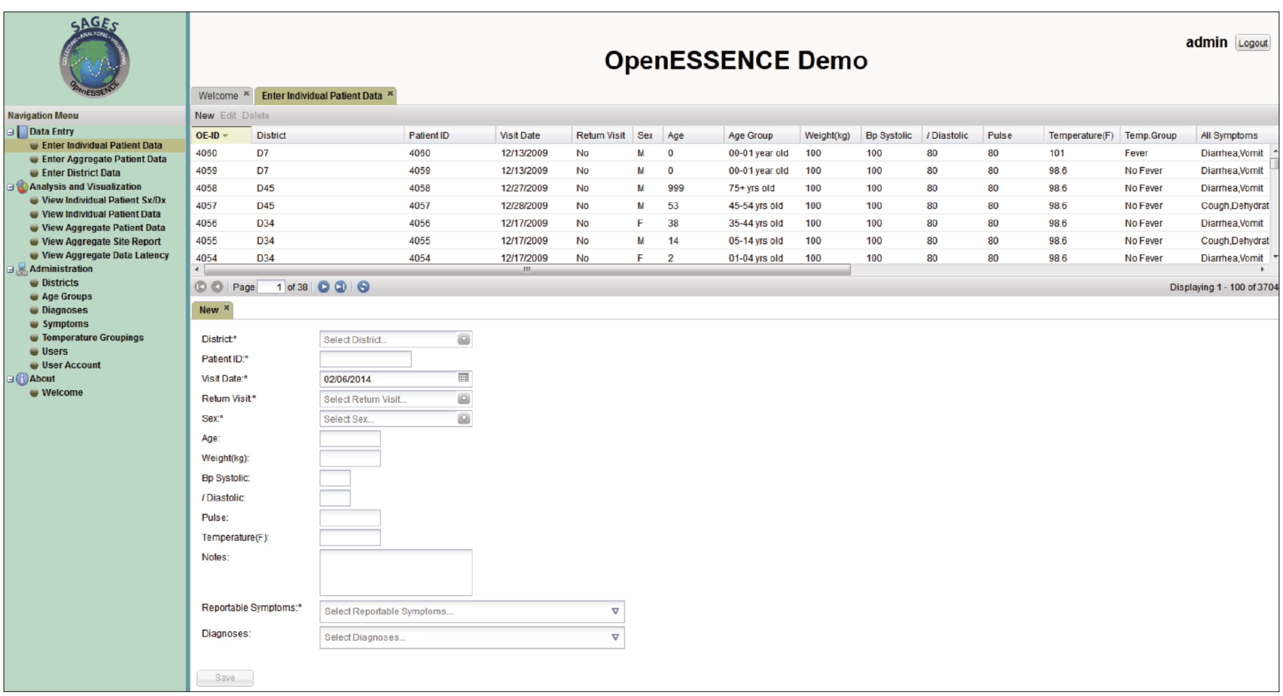


Figure 3. User interface for entering data into OE.

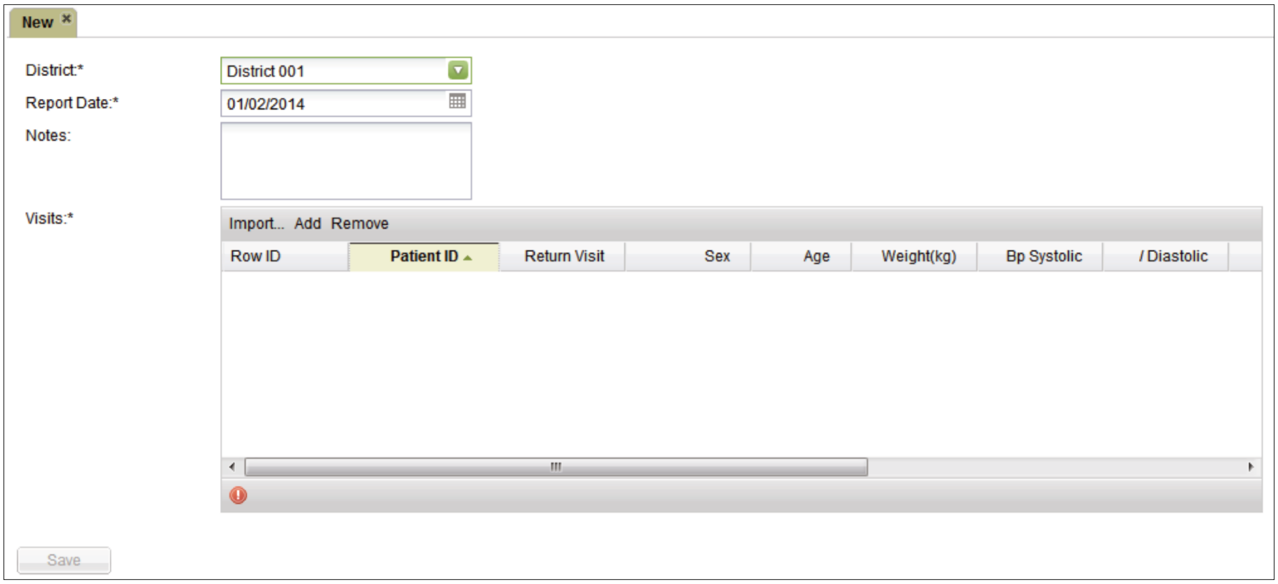


Figure 4. A screenshot illustrating the CSV file upload feature.

option in mCollect is via Hypertext Transfer Protocol (HTTP) if the Internet is available to the phone—forms are sent directly to the computer running OE. Secured messaging is also achieved with either 128-bit Advanced Encryption Standard[16] SMS message encryption or form data submission via Hypertext Transfer Protocol Secure (HTTPS).

In areas with Internet access, a Web-based user interface can be used. Figure 3 illustrates this interface for entering various types of data, including individual and aggregated patient data. The data can be entered by typing free text and numeric data, as well as by choosing different selections from pull-down lists. Pull-down lists support autocompletion, allowing the user to begin typing to find a selection more quickly. Free text data can be validated with configurable data parameters, such as maximum/minimum ranges, decimals, etc.

Visits are entered using a grid editor field that allows the user to add or remove an individual visit by clicking the “Add” or “Remove” button (Fig. 4). A new feature of OE is the ability to upload health data from a comma-separated value (CSV) file. This allows the user to upload a file containing a bulk quantity of any type of data, such as daily patient visits for a selected health district, all at one time.

The grid shown in Fig. 4 provides a CSV import feature allowing the user to select a CSV file location, as shown in Fig. 5. Figure 5 shows how the user can indicate the configuration of the CSV file, such as specifying inclusion of headers, its delimiter, and any qualifiers for the data of interest. Once these upload options are selected, the user can then preview the data to confirm the selections before uploading to the grid.

For data input through a database connection, standard extract, transform, and load synchronization processes can be used to load surveillance data from any Java Database Connectivity–compatible database. Users can configure the system specifically for the variables included in their database, thereby easing common data ingestion problems, especially the difficulties in trying to get disparate data formats to fit a specific type of data ingestion.

Analysis and Visualization

OE provides a variety of tools to analyze and visualize data and any derived data products. Furthermore, OE provides the ability for multiple users to share information derived from data without having to reveal health data that may be sensitive or private. These tools enhance the process of public health decision making by allowing visualization of disparate data types and analytic results and by facilitating the sharing of public health information across jurisdictional boundaries.

Queries are used to filter the data for analysis by date range, selection from list, multiselection, free form text/number, etc. The query form is shown in Fig. 6. The free-form text input fields support database wildcards. Combinations of logical “AND” and “OR” operators can

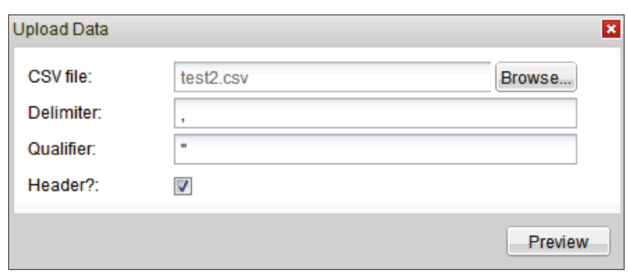


Figure 5. CSV upload window showing the selections that can be made by the user.

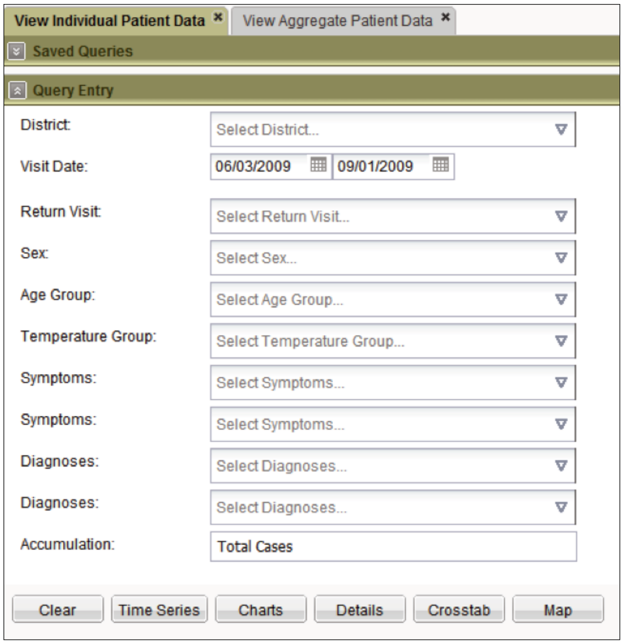


Figure 6. Screenshot of the data report query form for OE. Note that previously created queries can be saved for later use.

be used in queries. Queries are built using a graphical user interface to limit the need for detailed knowledge of structured query languages. This query builder approach focuses all database access logic into one part of the code base, thereby making it easier to manage. These queries may be saved and used to establish case definitions, allowing the user to repeat his or her analysis in a similar fashion and compare results.

Figure 7 is a screenshot showing how data can be analyzed by examining different categorizations of data relationships, such as sex, age group, district, and symptom. These features may allow the user to determine whether the disease outbreak is more prevalent within or among different data groups. Outbreak detection can be improved if the data sources contain information that can be used to include individuals potentially affected by an event and to exclude those unaffected.

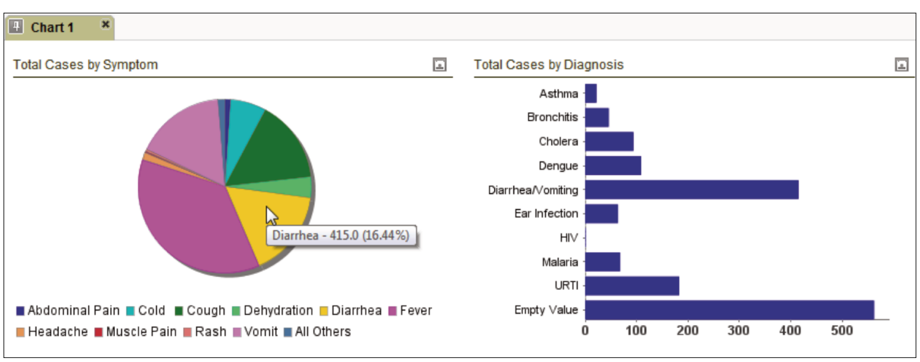


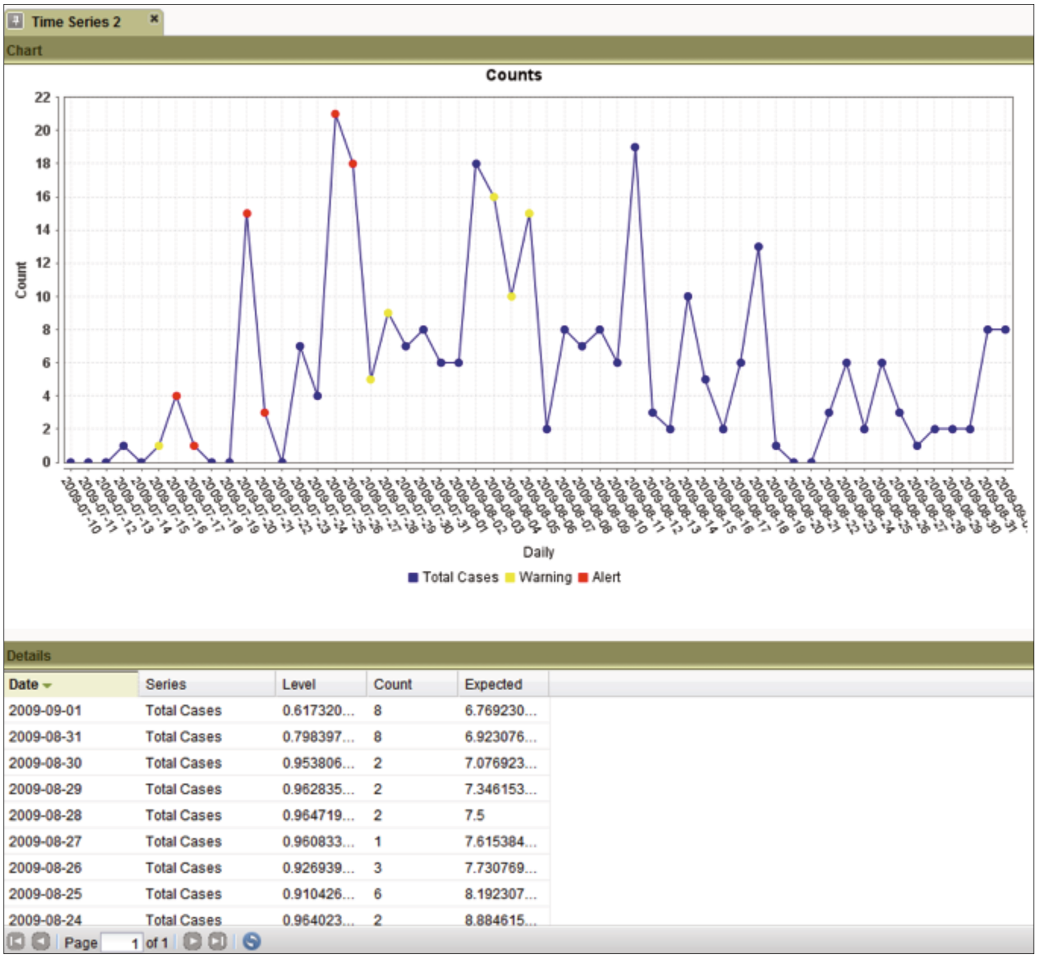
Figure 7. OE screenshot illustrating configurable charting, including pie and bar charts of configured groupings.

Figure 8 shows a screenshot of time series and detection analysis. Time series plots can now be arranged to display different years, one above the other, so that a comparison can be made to see what was happening at the same time in previous years, or whether the seasonal illness patterns are significantly different. Clicking a point on the time series plot shows the user a detailed view of the records. Giving the user the ability to drill deeply into the data allows the user to examine the specific health information that is resulting in an algorithm-derived alert. Data details, which preserve column ordering and sorting, can be exported to CSV files for external analysis using other tools such as Microsoft Excel. Time-series visualization includes anomaly detection and image export. These features assist users in sharing information while investigating a possible outbreak.

OE uses a plug-in application programming interface for detection algorithms that allows a developer to add additional detection algorithms. Currently supported detection algorithms include open-source algorithms such as Generalized Adaptive Smoothing, Cumulative Summation, and Exponentially Weighted Moving Average, in addition to different varieties of the U.S. Centers for Disease Control and Prevention Early Aberration Reporting System algorithms (EARS C1, C2, and C3). Creating a time series without running a detection algorithm is also supported.

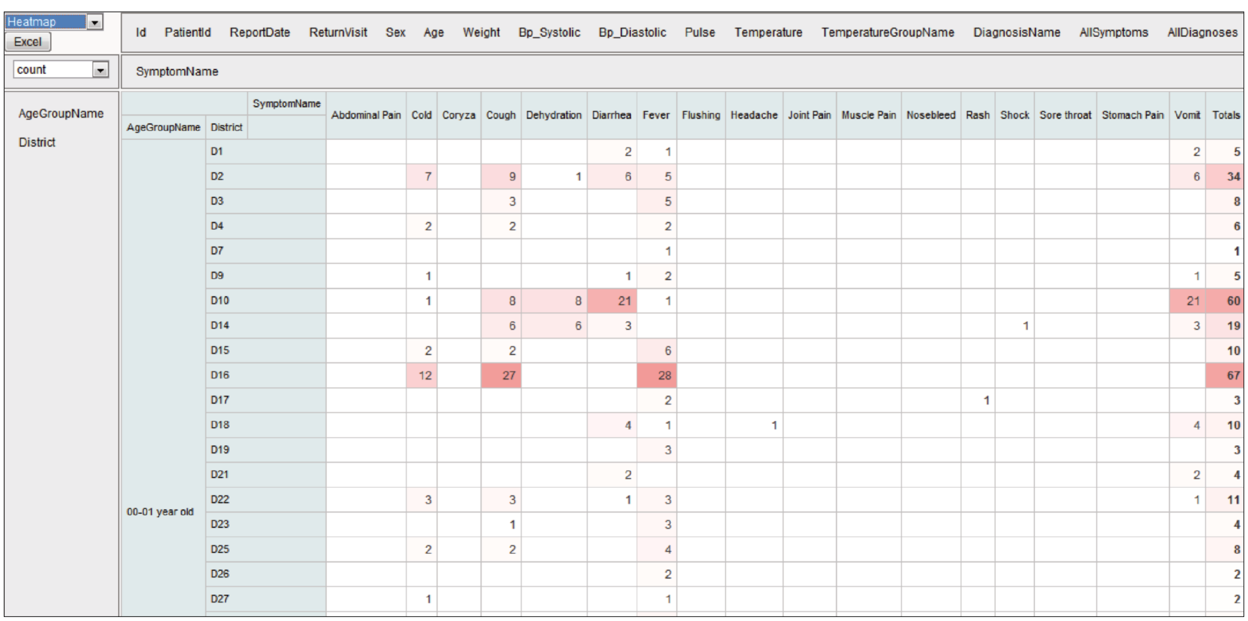
A new feature of OE allows the user to create a crosstab report (or N × N × N table) to display the frequency of data on the basis of multiple variables, such as district, age group, and symptom. This is similar to the concept of pivot tables in multidimensional reports. The crosstab report is dynamically generated from the data set, allowing users to select and update its dimensions for different comparisons. In addition, data filters can be applied when creating the crosstab to presample the data. The default crosstab is an *n*-dimensional matrix that shows the aggregated values of the selected comparison. The report also provides additional visualization features such as heat maps and table bar charts to aid in analysis of the comparison. An example of a crosstab using the heat map visualization is shown in Fig. 9. Crosstab data can also be exported to a CSV file for external analysis.

The output of different data queries and detection algorithms may be used to create maps with specific types of information. This facilitates quick identification of the geographic regions that are impacted and any notable



Date	Series	Level	Count	Expected
2009-09-01	Total Cases	0.617320...	8	6.769230...
2009-08-31	Total Cases	0.798397...	8	6.923076...
2009-08-30	Total Cases	0.953806...	2	7.076923...
2009-08-29	Total Cases	0.962835...	2	7.346153...
2009-08-28	Total Cases	0.964719...	2	7.5
2009-08-27	Total Cases	0.960833...	1	7.615384...
2009-08-26	Total Cases	0.926939...	3	7.730769...
2009-08-25	Total Cases	0.910426...	6	8.192307...
2009-08-24	Total Cases	0.964023...	2	8.884615...

Figure 8. OE screenshot of time series analysis. Below the plot are details of the detection algorithm output. Detection algorithm alerts are indicated by the red peaks in the plot.



AgeGroupName	District	Abdominal Pain	Cold	Coryza	Cough	Dehydration	Diarrhea	Fever	Flushing	Headache	Joint Pain	Muscle Pain	Nosebleed	Rash	Shock	Sore throat	Stomach Pain	Vomit	Totals
00-01 year old	D1						2	1										2	5
	D2		7		9	1	6	5										6	34
	D3				3			5											8
	D4		2		2			2											6
	D7							1											1
	D9		1				1	2										1	5
	D10		1		8	8	21	1										21	60
	D14				6	6	3								1			3	19
	D15		2		2			6											10
	D16		12		27			28											67
	D17							2						1					3
	D18						4	1		1								4	10
	D19							3											3
	D21						2											2	4
	D22		3		3		1	3										1	11
	D23				1			3											4
	D25		2		2			4											8
	D26							2											2
	D27		1					1											2

Figure 9. Example of a user-created crosstab employing the heat map visualization feature.

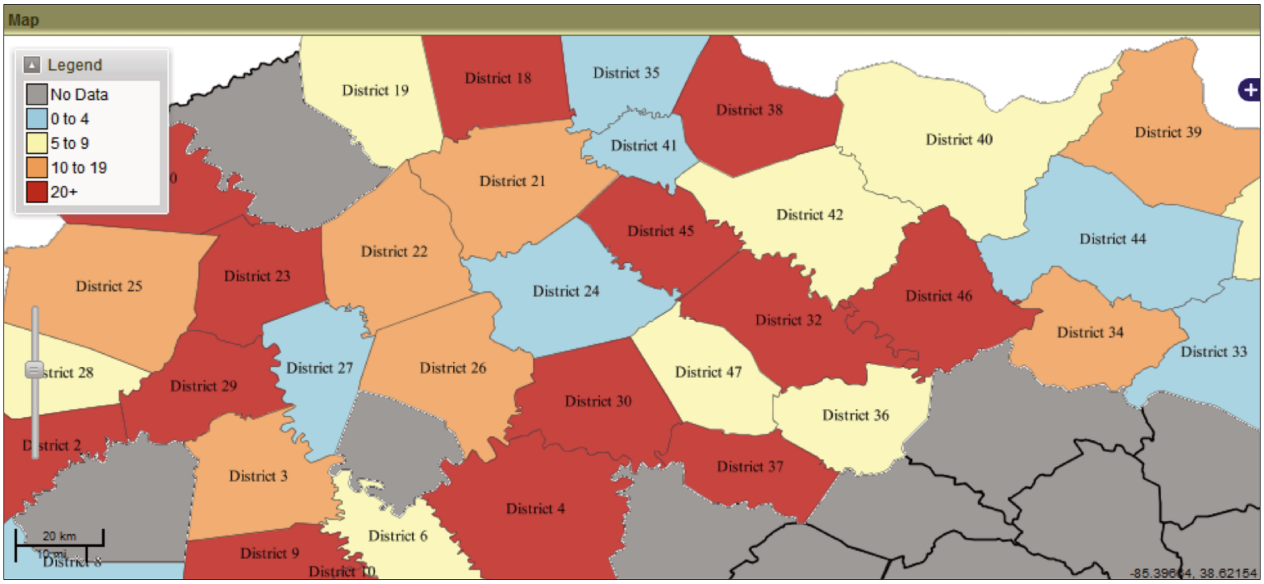


Figure 10. OE screenshot of geographic information system (GIS) map of data results.

outbreak patterns. Figure 10 shows OE analysis of data by geographic region. The ability of OE to export such geographic images may assist users in sharing information while investigating a possible outbreak. Although not incorporated into OE, there is the potential to use detection algorithms designed specifically for geographic data, such as SaTScan (http://www.satscan.org)[17]; such algorithms analyze whether a disease is occurring in clusters or is randomly distributed.

CONCLUSIONS

By gathering information from public health stakeholders around the world, including those in resource-limited countries, APL created OE to meet the following user-defined requirements for a globally deployed biosurveillance system: (*i*) the ability to maintain control of one's own data is very important to the users; (*ii*) no cost/no recurring fee open-source software is particularly desirable; (*iii*) the system should place a minimum burden on those providing data; (*iv*) system acceptance results from enabling the user to easily tailor the system to local needs; and (*v*) sustainability results from local ownership and working within existing needs and resources. Therefore, OE offers self-contained disease surveillance tools that can be deployed efficiently at a variety of resource-limited, as well as disaster, locations. As an open-source application, OE is freely available and easily deployable, upgradeable, and extendable.

OE was developed to support a wide variety of user needs in different settings, including user-defined preferences and mechanisms for data input from several types of databases. The system structure is data driven to allow for dynamic extension and reconfiguration. Users can design their own queries and use different detection algorithms to analyze their data. OE provides the user with results in a format similar to that provided by Enterprise ESSENCE, including graphs, charts, detailed data on individual illness reports, and geographic maps of locations of individual illness reports.[4] The OE system can also be used to share actionable information via the Internet among multiple users and across different jurisdictions. OE can be used globally to improve the timeliness of health data collection, enhance the early detection of disease outbreaks, and improve situational awareness.

ACKNOWLEDGMENTS: We acknowledge the collaboration with and financial support of AFHSC-GEIS. We have received invaluable assistance from our colleagues at several U.S. DoD overseas laboratories as well as from our foreign military and civilian colleagues at the many SAGES installation sites. We thank Shradda Patel and Brian Feighner for their assistance with this article. We are deeply indebted to Wayne Loschen, the developer of the Enterprise ESSENCE system, for his sage guidance on these projects.

REFERENCES

[1]Jones, K., Patel, N., Levy, M., Storeygard, A., Balk, D., et al., "Global Trends in Emerging Infectious Diseases," *Nature* **451**(7181), 990–993 (2008).

[2]Katz, R. L., Gornto, S. E., and Chretien, J. P., "Expert Opinion on Implementation Strategies for the International Health Regulations," *World Med. Health Policy* **3**(3), 1–12 (2011).

[3]Ashar, R., Lewis, S., Blazes, D., and Chretien, J., "Applying Information and Communications Technologies to Collect Health Data from Remote Settings: a Systematic Assessment of Current Technologies," *J. Biomed. Inform.* **43**(2), 332–341 (2010).

[4]Lombardo, J., Burkom, H., Elbert, E., Magruder, S., Happel-Lewis, S., et al., "A Systems Overview of the Electronic Surveillance System for the Early Notification of Community-Based Epidemics (ESSENCE II)," *J. Urban Health* **80**(2 Suppl 1), i31–i41 (2003).

[5]Lucero, C., Oda, G., Cox, K., Maldonado, F., Lombardo, J., et al., "Enhanced Health Event Detection and Influenza Surveillance Using a Joint Veterans Affairs and Department of Defense Biosurveillance System," *BMC Med. Inform. Decis. Mak.* **11**(19 Sep), 56 (2011).
[6]Lewis, S., Feighner, B., Loschen, W., Wojcik, R., Skora, J., et al., "SAGES: A Suite of Freely Available Software Tools for Electronic Disease Surveillance in Resource-limited Settings," *PLoS One* **6**(5), e19750 (2011).
[7]Lewis, S., Feighner, B., Campbell, T., Wojcik, R., Coberly, J., and Blazes, D., "Suite for Automated Global Electronic bioSurveillance (SAGES)," in *Proc. U.S. Centers for Disease Control and Prevention 2011 Public Health Informatics Conf.*, Atlanta, GA (2011).
[8]Kalra, D., and Forslund, D., "Open Source Health Systems," *Person-Centered Health Records*, J. Demetriades, R. Kolodner, and G. Chrsitopherson (eds.), Springer, New York, pp. 169–185 (2005).
[9]Campbell, T., Hodanics, C., Babin, S., Poku, A., Wojcik, R., et al., "Developing Open Source, Self-Contained Disease Surveillance Software Applications for Use in Resource-Limited Settings," *BMC Med. Inform. Decis. Mak.* **12**(6 Sep), 99 (2012).
[10]Johnson, R., Hoeller, J., Arendsen, A., and Risberg, T., *Professional Java Development with the Spring Framework*, Wrox Press, New York (2005).
[11]Koenig, D., Glover, A., King, P., Laforge, G., Skeet, J., and Gosling, J., *Groovy in Action*, Manning Publications, Shelter Island, New York (2007).
[12]Babin, S., Magruder, S., Hakre, S., Coberly, J., and Lombardo, J., "Understanding the Data: Health Indicators in Disease Surveillance," *Disease Surveillance: A Public Health Informatics Approach*, J. Lombardo and D. Buckeridge (eds.), John Wiley and Sons, Hoboken, New Jersey, pp. 43–90 (2007).
[13]Lewis, S., and Wojcik, R., "Methodologies for Data Collection," *BMC Proc.* **2**(Suppl 3), S5 (2008).
[14]Coberly, J., Tomayo, A., Velasco, J. M., Cabugao, F., Tac-an, I., and Wojcik, R., "Dengue SMS Surveillance Project in the Philippines," in *Proc. U.S. Centers for Disease Control and Prevention Public Health Information Network Conf.*, Atlanta, GA (2009), https://cdc.confex.com/cdc/phin2009/webprogram/Paper20977.html.
[15]The Mobile Broadband Standard: A Global Initiative, *3rd Generation Partnership Project (3GPP) Specification 03.40 for the Technical Realization of the Short Message Service (SMS)*, http://www.3gpp.org/DynaReport/0340.htm (accessed 9 Jan 2014).
[16]Paar, C., and Pelzl, J., "The Advanced Encryption Standard, (AES)" Chap. 4, *Understanding Cryptography: A Textbook for Students and Practitioners*, Springer, New York (2009).
[17]Burkom, H., "Biosurveillance Applying Scan Statistics with Multiple, Disparate Data Sources," *J. Urban Health* **80**(2 Suppl 1), i57–i65 (2003).

The Authors

Timothy C. Campbell was the SAGES technical lead and developer for the OE application. **Charles J. Hodanics**, **Zarna S. Mistry**, **Gabriel N. Gorelick-Feldman**, and **Colin J. Taylor** designed and developed features for the OE application. **Adjoa M. Poku** was the technical lead for telephone and SMS data ingestion design and development and was a software developer for OE. **Richard A. Wojcik** was the Chief Engineer for all versions of ESSENCE. **Steven M. Babin** was the subject-matter expert in medicine and public health. **Jacqueline S. Coberly** was the Lead Epidemiologist on the project and obtained user input and usage data on the OE application. **Aaron T. Katz** was the Project Manager for OE. **Sheri H. Lewis** is the Program Manager for Global Health Surveillance. For further information on the work reported here, contact Timothy Campbell. His e-mail address is timothy.campbell@jhuapl.edu.

The *Johns Hopkins APL Technical Digest* can be accessed electronically at **www.jhuapl.edu/techdigest**.