OpenESSENCE User Guide

For the OpenESSENCE demo site

Prepared by Johns Hopkins Applied Physics Laboratory,
for
Armed Forces Health Surveillance Center, Division of GEIS Operations

July 2014

Table of Contents

INTRODUCTION

History of ESSENCE and SAGES

ESSENCE is the acronym for the Electronic Surveillance System for the Early Notification of Community-based Epidemics. It is a web-based system used by state and city public health authorities throughout the United States to monitor disease trends in their communities. The system was conceived to help public health authorities rapidly identify disease outbreaks possibly associated with bio-terrorist attacks. Epidemiologists using the system quickly determined, however, that the technology was as useful—or more useful—in monitoring trends and outbreaks of naturally occurring diseases. The enterprise ESSENCE system requires a high speed internet connection, relies on automated data streams, and uses proprietary software for the display of data. It became clear over time that an ESSENCE-type system would be useful in geographic areas and situations where the existing technology infrastructure is limited and cannot support enterprise ESSENCE.

SAGES (Suite for Automated Global Electronic bioSurveillance) leverages the experience gained in the development of ESSENCE by the US Department of Defense and the Johns Hopkins University Applied Physics Laboratory (JHU/APL), and applies that experience to enhancing disease surveillance capacity in resource-limited settings around the world.

SAGES tools are organized into four categories: 1) data collection, 2) analysis and visualization, 3) communications, and 4) modeling/simulation/evaluation (Figure 1). Within each category, SAGES offers a variety of tools compatible with surveillance needs and different types or levels of information technology infrastructure.

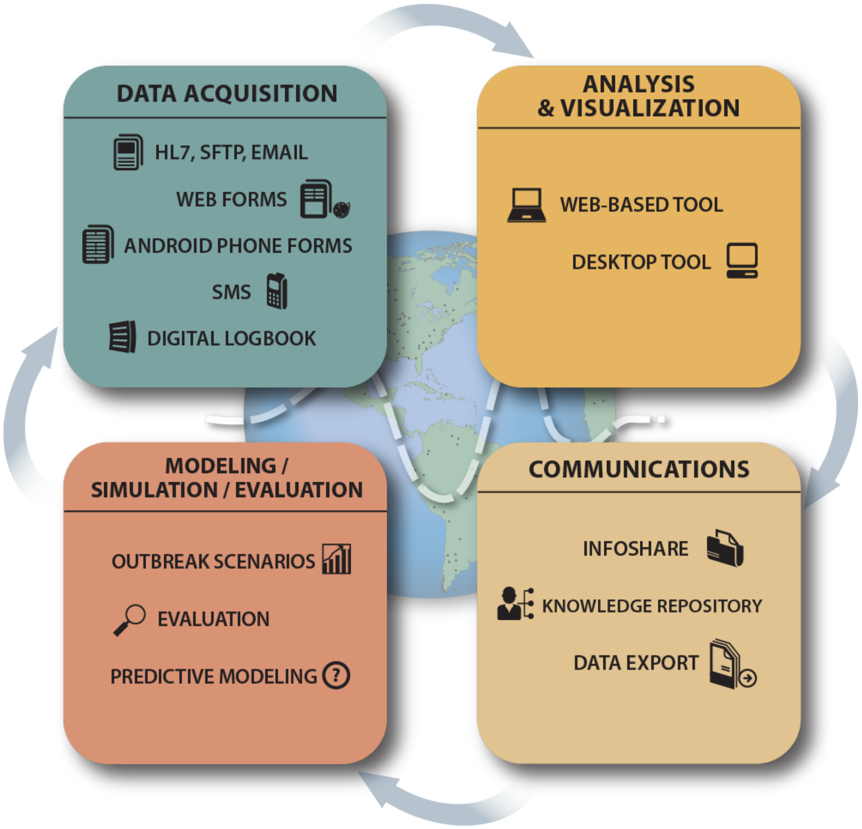


Figure 1: SAGES Suite of Tools

OpenESSENCE is a part of the SAGES suite of tools. It is a freely-available, web-based, multi-user, data entry, analysis, and visualization tool for electronic disease surveillance. Although it includes many of the features and much of the functionality of the more mature enterprise ESSENCE system, it does not require a high speed internet connection or automated data streams. Features include demographic characterization tools, temporal and spatial analysis, patient level information display, geographic information system mapping, anomalous event detection, and dynamic query capability.

SAGES/OpenESSENCE System Architecture Example

In resource-limited settings, it is imperative to select technology that is both easy to incorporate into existing health services and sustainable with little or no additional financial investment. Each implementation of SAGES is designed to fit the needs of the environment into which it is being installed, and to use the existing infrastructure. The system can be scaled up to meet changing needs and increased availability of resources.

There are five basic processes in the SAGES framework (Figure 2). Within each process, several different component options are available.

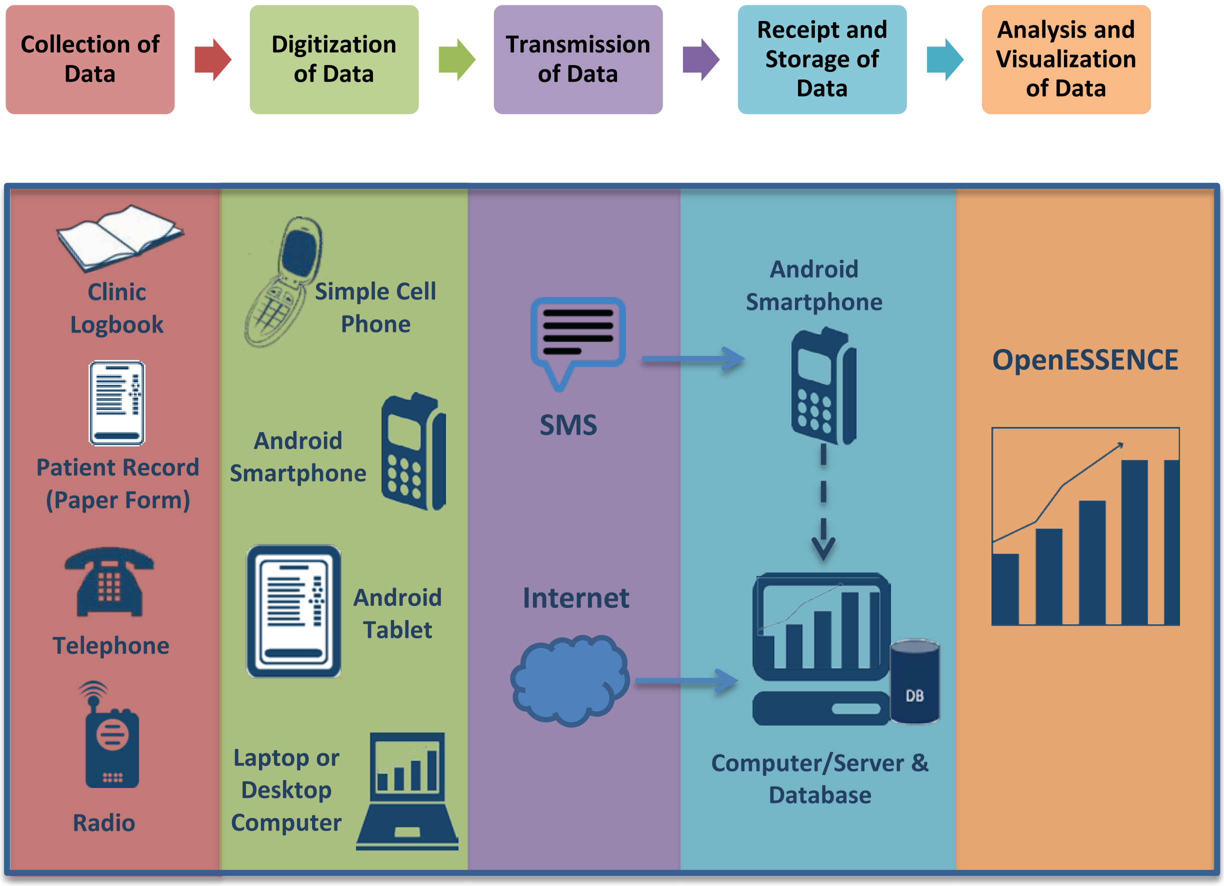


Figure 2: SAGES Processes and Components

OpenESSENCE is a web-based tool. The basic OpenESSENCE system architecture includes a central server (depending upon the data volume, the machine can be an actual server class computer, a desktop, or laptop) which hosts the database and the OpenESSENCE web application. Any computer with an internet connection can log in to the OpenESSENCE server using a web browser, and perform data analysis and data entry.

A basic SAGES system architecture example is shown in Figure 3. One or more patient care sites can be incorporated into this architecture, creating a regional or national surveillance architecture.

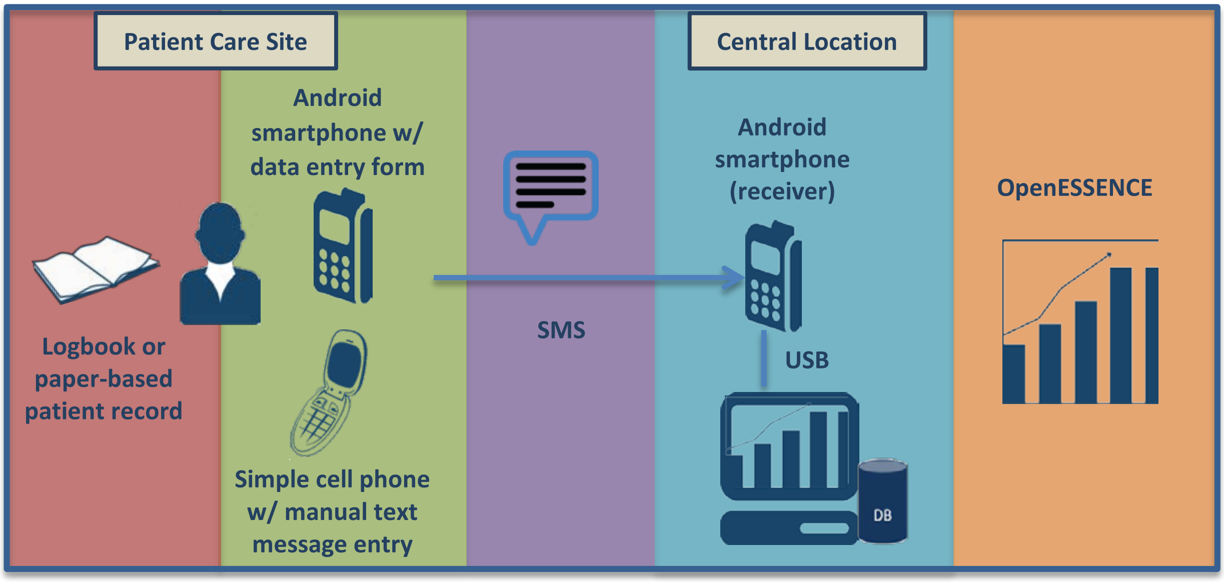

Patient Care Site
Central Location
Android smartphone w/ data entry form
Logbook or paper-based patient record
Simple cell phone w/ manual text message entry
SMS
Android smartphone (receiver)
USB
DB
OpenESSENCE

Figure 3: SAGES System Architecture Example

At the patient care site (such as a clinic or hospital), information from existing paper records or logbook entries is digitized by manually entering the information into Short Message Service (SMS) text messages using simple cell phones or Android smartphones. A data collection form can be used on the Android smart phone for ease of data entry; if using a simple cell phone, the SMS text messages would adhere to a pre-defined format. Having the option to use a smartphone or a simple cell phone illustrates the flexibility of the SAGES framework. Smartphones are a higher cost resource that may not, initially, be available at all locations. As additional resources become available (tablets, computers, internet service), they can be incorporated into the architecture.

The SMS messages are received at a central point on an Android smartphone, which is attached via USB to a computer running OpenEssence. The SMS messages are automatically pulled from the Android smartphone through a scheduled batch process that parses and writes the data into the OpenESSENCE database. The data is now ready to be analyzed using the analysis and visualization capabilities of OpenESSENCE. The OpenESSENCE web application is accessible by any computer that can connect to the central OpenESSENCE web server over the internet/intranet.

If a computer with internet access is available at the patient care site, data can be entered directly into OpenESSENCE (Figure 4). Data analysis can also be performed from that computer (time series, charts, maps).

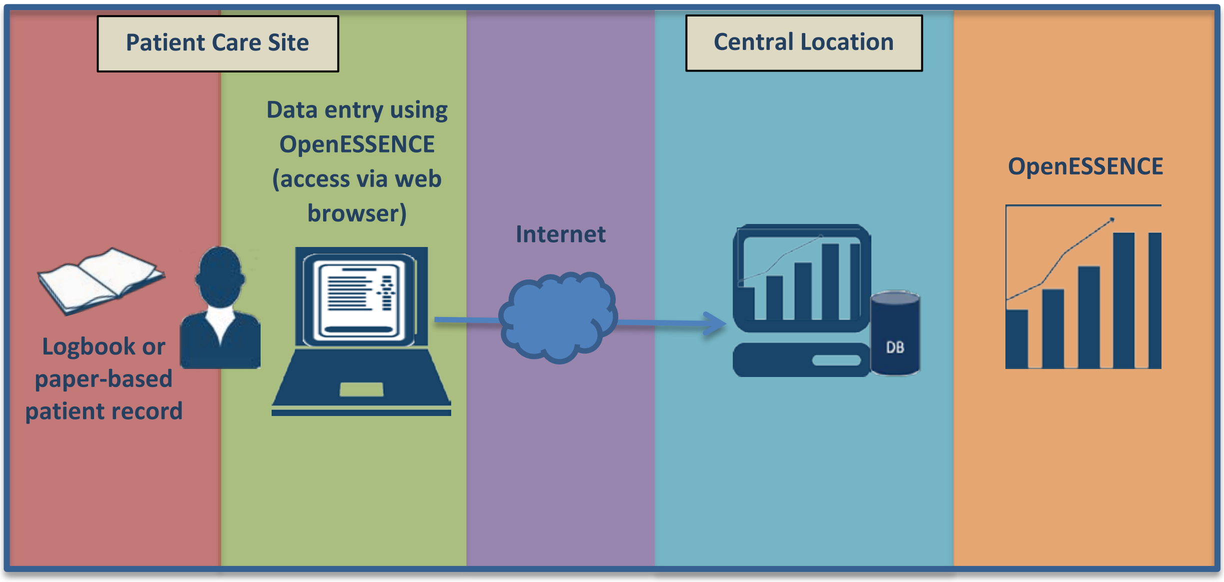

Patient Care Site
Central Location
Data entry using OpenESSENCE (access via web browser)
Internet
OpenESSENCE
Logbook or paper-based patient record
DB

Figure 4: SAGES System Architecture Example with Computer and Internet at Patient Care Site

GETTING STARTED

OpenESSENCE is highly flexible and customizable. Components such as data elements, data types, record types, column names, language and font are all customizable. Each implementation of the system will be unique. The best way to start learning about the tool is to use the OpenESSENCE demonstration (or 'demo') site, which is hosted by JHU/APL. This site contains sample data and also supports entry of both cased-based and aggregate data. All of the OpenESSENCE analysis and visualization tools are available on the demo site. The site is meant to be a representation of an actual instance of OpenESSENCE that would be implemented in a country, region, or geographic area.

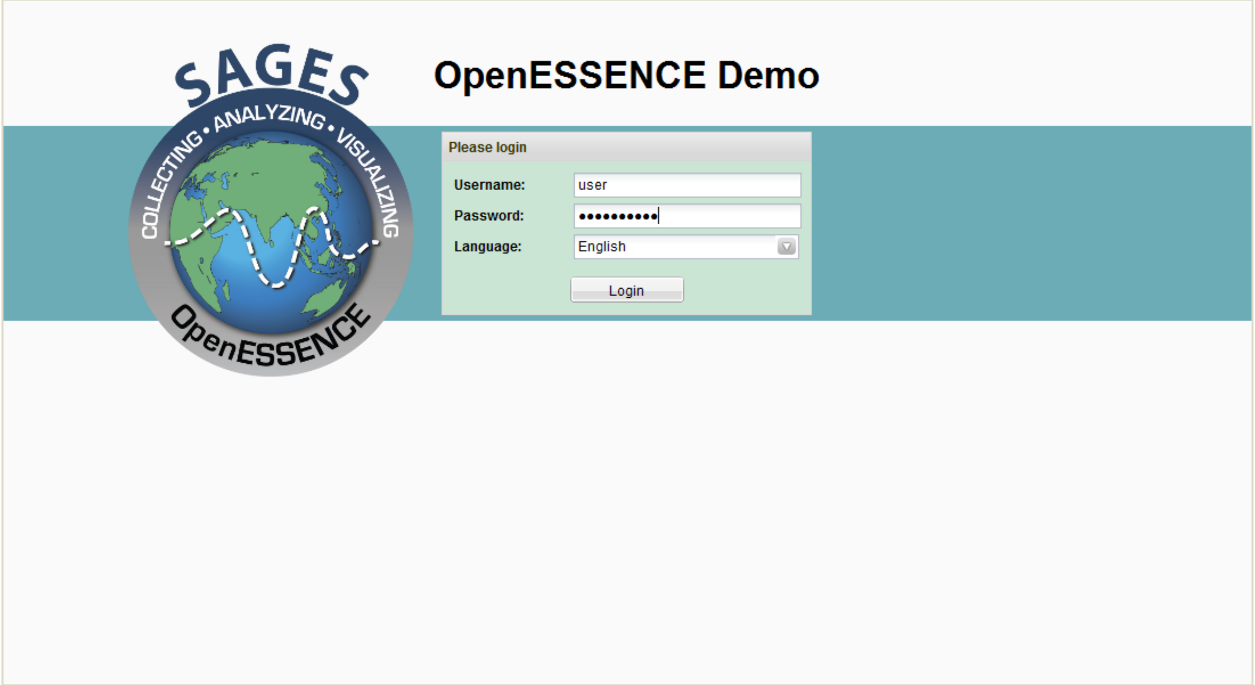
This User Guide is based on the OpenESSENCE demo site. An actual implementation of OpenESSENCE will likely be different from the demo site. However, the data entry, analysis, and visualization features and concepts remain the same.

Log in to the OpenEssence Demo Site

1. Using a web browser, go to the following web address: https://128.244.178.159/openessence
2. In the **Username** field, type the username.
3. In the **Password** field, type the password.

   NOTE: The username and password will be provided to you separately.

4. OPTIONAL: Select the appropriate language from the **Select Language** list.
5. Click **Login**.



OpenESSENCE Demo

Please login

Username: user

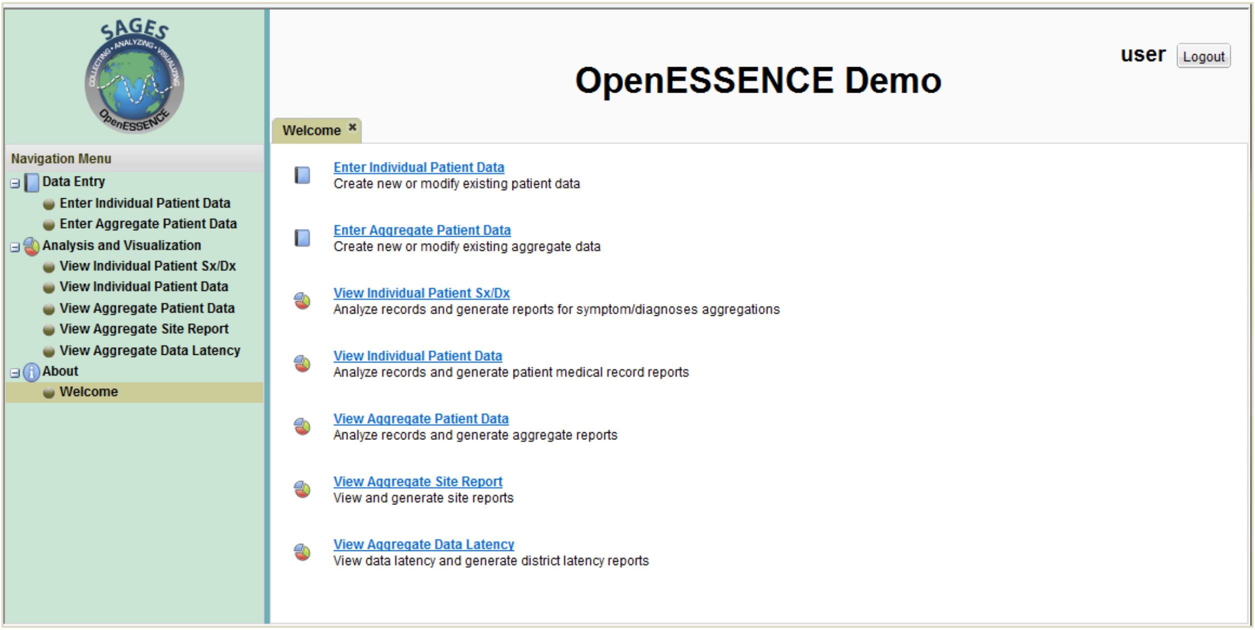
Password: ••••••••••

Language: English

A typical OpenESSENCE installation has two types of accounts or user roles (a general user and an administrator). For example, a user account can provide data entry, modification, deletion, and analysis capability; or, a user account can be configured to allow only data entry capability. Multiple accounts can be created, so that each user has a unique user name and password. An administrator account allows for maintenance of reference tables and creation of additional user accounts and is generally limited to personnel who maintain the OpenESSENCE system.

Navigation Menu

The initial OpenESSENCE window contains a **Navigation Menu** on the left, and a **Welcome** tab which is open by default. The **Welcome** tab can be closed by clicking the **X** next to the tab name. At the top right corner of the window, the name of the user currently logged in is displayed (in our example, the user name is '*user*').


SAGES
COLLECTING • ANALYZING • VISUALIZING
OpenESSENCE
Navigation Menu
Data Entry
Enter Individual Patient Data
Enter Aggregate Patient Data
Analysis and Visualization
View Individual Patient Sx/Dx
View Individual Patient Data
View Aggregate Patient Data
View Aggregate Site Report
View Aggregate Data Latency
About
Welcome
user
Logout
Welcome ×

OpenESSENCE Demo

Enter Individual Patient Data
Create new or modify existing patient data

Enter Aggregate Patient Data
Create new or modify existing aggregate data

View Individual Patient Sx/Dx
Analyze records and generate reports for symptom/diagnoses aggregations

View Individual Patient Data
Analyze records and generate patient medical record reports

View Aggregate Patient Data
Analyze records and generate aggregate reports

View Aggregate Site Report
View and generate site reports

View Aggregate Data Latency
View data latency and generate district latency reports

The Navigation Menu provides access to the data entry and analysis functions in OpenESSENCE. New tabs will open in the center pane when items are selected from the Navigation Menu.

The **Data Entry** options allow for the entry of individual patient data and aggregate patient data. The following options are available for data entry:

- **Enter Individual Patient Data** – enter, modify, or delete case-based data records
- **Enter Aggregate Patient Data** – enter, modify, or delete aggregate data records

The **Analysis and Visualization** options enable the use to analyze and generate reports using time series graphs, pie/bar charts, geographic maps, and tabular grids. The following options are available for analysis and visualization:

- **View Individual Patient Sx/Dx** – view selected individual patient records by symptom and/or disease (pie/bar charts, crosstabs, and details in tabular form only)
- **View Individual Patient Data** – view selected individual patient data, using time series graphs, pie/bar charts, crosstabs, geographic mapping, or details in tabular form

- **View Aggregate Patient Data** – view selected aggregate patient data by symptom(s) accumulation, using time series graphs, pie/bar charts, crosstabs, or details in tabular form
- **View Aggregate Data Latency** – view reporting latency by district, using time series graphs, pie/bar charts, crosstabs, or details in tabular form
- **View Aggregate Site Report** – view the number of aggregate patient data reports that have been created by district and date, using time series graphs, pie/bar charts, crosstabs, or details in tabular form

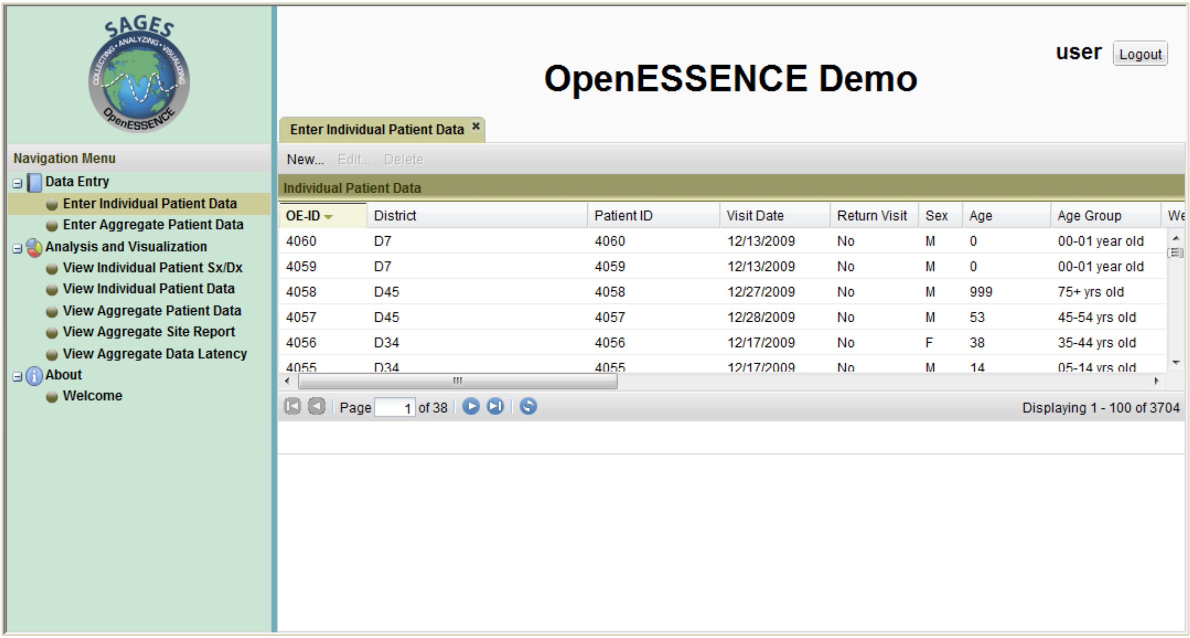
These options are explained in detail in the subsequent sections of this User Guide.

DATA ENTRY

The OpenESSENCE Demo site supports two input data types: Individual Patient Data and Aggregate Patient Data. The Individual Patient Data input type lets you enter data for each unique patient visit; the Aggregate Patient Data input type lets you enter aggregate data, or the number of patients with a particular complaint or symptom during a pre-defined time period.

Create a New Individual Patient Data Record

1. On the OpenESSENCE Demo screen, locate the **Navigation Menu** on the left.
2. Under **Data Entry**, select **Enter Individual Patient Data** (case-based). A new tab called **Enter Individual Patient Data** appears on the right hand side of the OpenESSENCE Demo window.

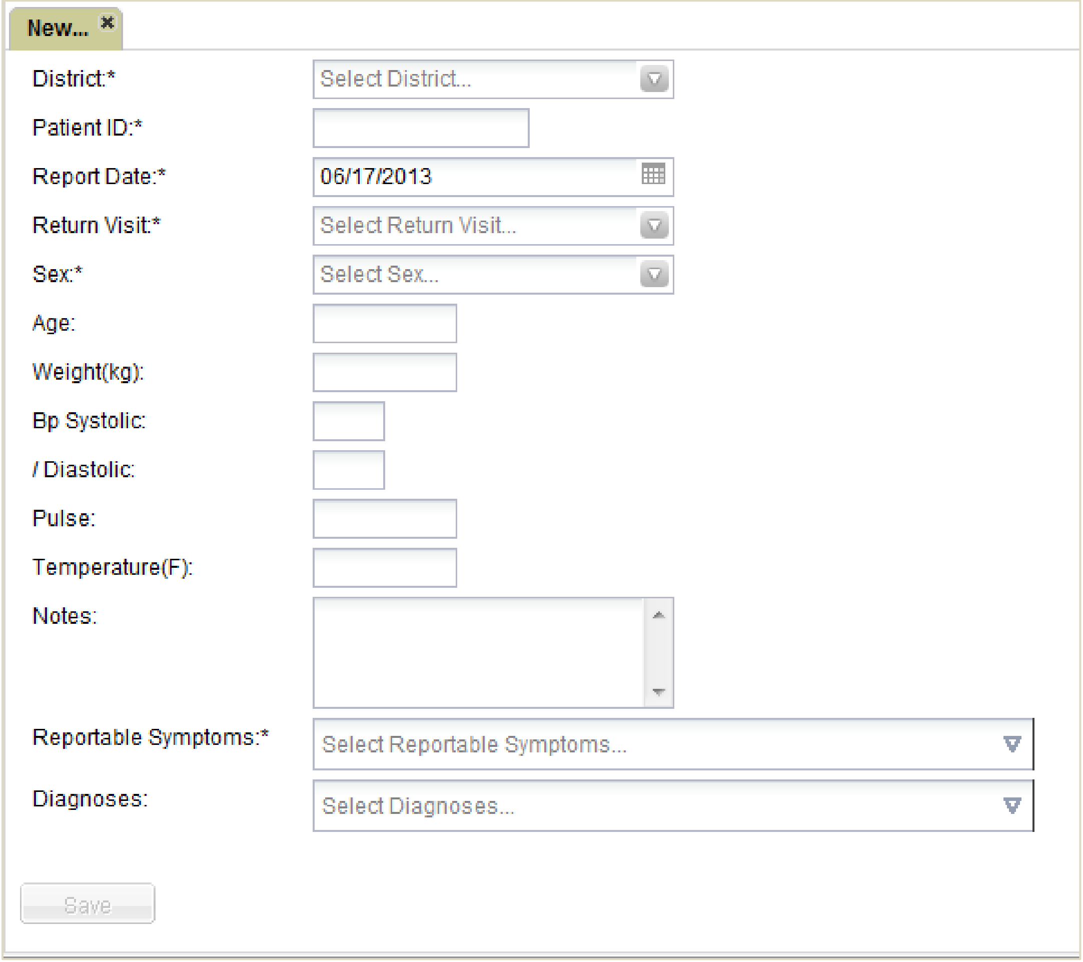


SAGES
COLLECTING • ANALYZING • VISUALIZING
OpenESSENCE
OpenESSENCE Demo
user
Logout
Navigation Menu
Data Entry
Enter Individual Patient Data
Enter Aggregate Patient Data
Analysis and Visualization
View Individual Patient Sx/Dx
View Individual Patient Data
View Aggregate Patient Data
View Aggregate Site Report
View Aggregate Data Latency
About
Welcome
Enter Individual Patient Data ×
New... Edit... Delete
Individual Patient Data

OE-ID	District	Patient ID	Visit Date	Return Visit	Sex	Age	Age Group	We
4060	D7	4060	12/13/2009	No	M	0	00-01 year old	
4059	D7	4059	12/13/2009	No	M	0	00-01 year old	
4058	D45	4058	12/27/2009	No	M	999	75+ yrs old	
4057	D45	4057	12/28/2009	No	M	53	45-54 yrs old	
4056	D34	4056	12/17/2009	No	F	38	35-44 yrs old	
4055	D34	4055	12/17/2009	No	M	14	05-14 yrs old	

Page 1 of 38
Displaying 1 - 100 of 3704

3. From the toolbar in the **Individual Patient Data** tab, click **New**. A new tab will appear below the list of existing Individual Patient Data records.
4. Fill in the appropriate information in the data fields. Required fields are marked with an asterisk (fields are described below).
5. Click **Save** to save the new Individual Patient record. The **Save** button is disabled until all required fields are filled in.



New... ✖

District:*	Select District...
Patient ID:*	
Report Date:*	06/17/2013
Return Visit:*	Select Return Visit...
Sex:*	Select Sex...
Age:	
Weight(kg):	
Bp Systolic:	
/ Diastolic:	
Pulse:	
Temperature(F):	
Notes:	
Reportable Symptoms:*	Select Reportable Symptoms...
Diagnoses:	Select Diagnoses...

District (required)	Geographic location of patient; geographic areas on the demo site are called districts; use pull-down list to select
Patient ID (required)	Unique ID for the patient; manual text entry (alphanumeric)
Report Date (required)	Date of report (default is current date; click calendar icon to select date)
Return Visit (required)	Indicate if visit is a return visit (Y/N)
Sex (required)	Select sex of patient (Female/Male/Unknown)
Age	Age of patient in years; manual text entry (numeric)
Weight (kg)	Weight of patient in kilograms; manual text entry (numeric)
Bp Systolic	Systolic blood pressure; manual text entry (numeric)

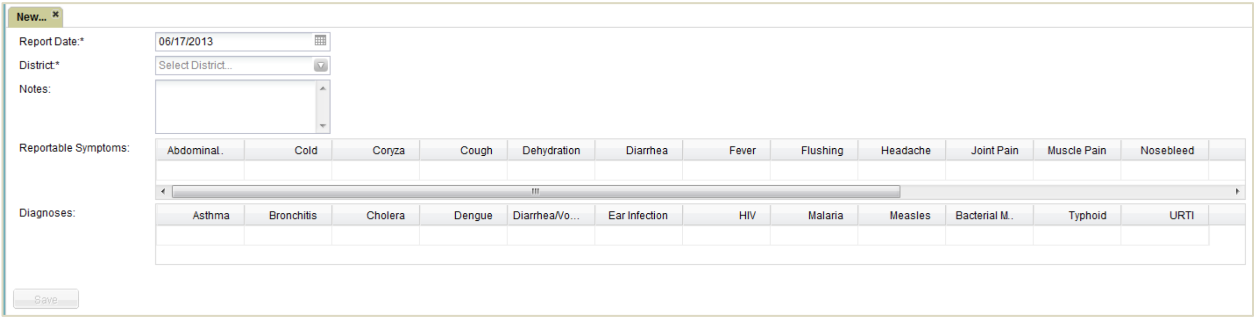
Diastolic	Diastolic blood pressure; manual text entry (numeric)
Pulse	Pulse; manual text entry (numeric)
Temperature (F)	Temperature in degrees Fahrenheit; manual text entry (numeric)
Notes	Additional notes as needed; manual text entry (alphanumeric)
Reportable Symptoms (required)	Symptom(s); select one or more from pull-down list
Diagnoses	Diagnoses; select one or more from pull-down list

Create a New Aggregate Patient Data Record

1. In the OpenESSENCE Demo window, locate the **Navigation Menu** on the left.
2. Under **Data Entry**, select **Enter Aggregate Patient Data**. A new tab called **Enter Aggregate Patient Data** appears on the right side of the Demo window.
3. From the toolbar in the **Enter Aggregate Patient Data** tab, click **New**. A new tab will appear below the list of existing Aggregate Patient Data records.
4. Fill in the appropriate information in the data fields. Required fields are marked with an asterisk (fields are described below). For **Reportable Symptoms** and **Diagnoses**, enter the count for each symptom or diagnosis as necessary.

   NOTE: Use the horizontal scroll bar to see the complete list of reportable symptoms and diagnoses.

5. Click **Save** to save the new Aggregate Patient Data record. The **Save** button is disabled until all required fields are filled in.



Report Date:* 06/17/2013

District:* Select District...

Notes:

Reportable Symptoms:

Abdominal.	Cold	Coryza	Cough	Dehydration	Diarrhea	Fever	Flushing	Headache	Joint Pain	Muscle Pain	Nosebleed

Diagnoses:

Asthma	Bronchitis	Cholera	Dengue	Diarrhea/Vo...	Ear Infection	HIV	Malaria	Measles	Bacterial M..	Typhoid	URTI

Report Date (required) — Date of report (default is current date; click calendar icon to select date)

District (required) — Geographic location of patient; geographic areas on the demo site are called districts; use pull-down list to select

Notes — Additional notes as needed; manual text entry (alphanumeric)

Reportable Symptoms — Enter count (numeric) under the relevant symptom(s)

Diagnoses — Enter count (numeric) under the relevant diagnoses

Edit an Existing Record

1. From the **Enter Individual Patient Data** tab or the **Enter Aggregate Patient Data** tab, select the record from the list of existing records.
2. From the toolbar, click **Edit**. A new tab will appear below the list of existing records, showing the unique record identifier for that record.
3. Modify the data fields as needed.
4. Click **Save** to save the changes to the record.

NOTE: You can select multiple records at one time for editing. Use the Shift key to select sequential records, or the CTRL key to select multiple non-sequential records.

Delete an Existing Record

1. From the **Enter Individual Patient Data** tab or the **Enter Aggregate Patient Data** tab, select the record from the list of existing records.
2. From the toolbar, click **Delete**.
3. A confirmation window appears. Click **Yes** to delete the record, or click **No** to cancel deletion.

NOTE: You can select multiple records at one time for editing. Use the Shift key to select sequential records, or the CTRL key to select multiple non-sequential records.

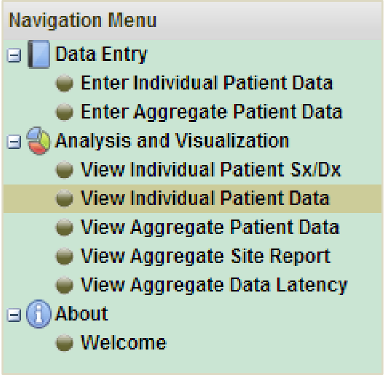
ANALYSIS AND VISUALIZATION

The **Analysis and Visualization** option allows the user to analyze and visualize the data by creating queries for data from the database. The queries are performed using query forms containing data fields which are used as filters to define the subset of data that you want to analyze. The results of the query can then be used to create time series graphs, charts, maps, crosstab reports, and tabular data.

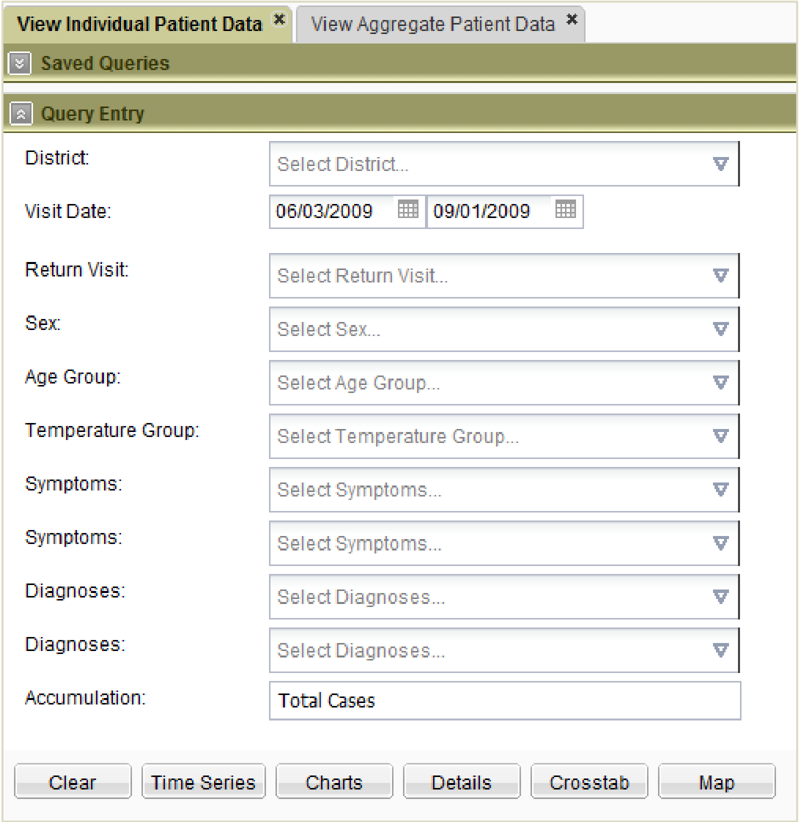
Create a Time Series Chart

The OpenESSENCE time series visualization feature displays data by aggregating and plotting data on a line chart. The tool also has the capability to run detection algorithms on the plotted data. This example will describe how to create time series charts for Individual Patient Data and Aggregate Patient Data.

1. To generate a time series chart, select either **View Individual Patient Data** or **View Aggregate Patient Data** from the **Navigation Menu**.


Navigation Menu
Data Entry
Enter Individual Patient Data
Enter Aggregate Patient Data
Analysis and Visualization
View Individual Patient Sx/Dx
View Individual Patient Data
View Aggregate Patient Data
View Aggregate Site Report
View Aggregate Data Latency
About
Welcome

2. A new tab corresponding to the data type will appear. The tab contains a query form containing data fields which are used as filters for time series charts, pie/bar charts, crosstabs, and maps.



District: Select District...

Visit Date: 06/03/2009 09/01/2009

Return Visit: Select Return Visit...

Sex: Select Sex...

Age Group: Select Age Group...

Temperature Group: Select Temperature Group...

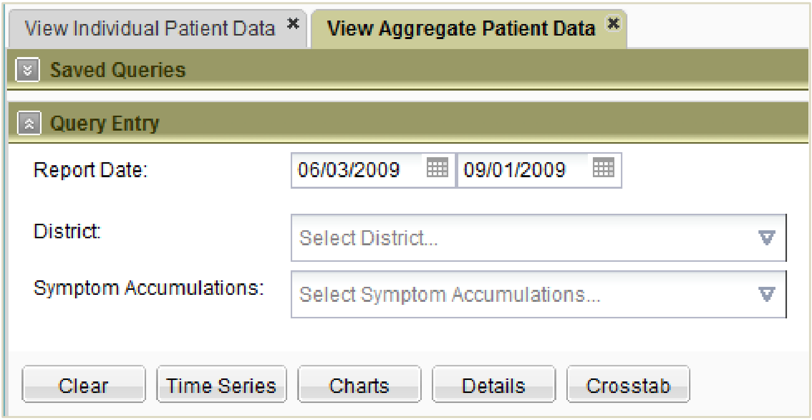
Symptoms: Select Symptoms...

Symptoms: Select Symptoms...

Diagnoses: Select Diagnoses...

Diagnoses: Select Diagnoses...

Accumulation: Total Cases

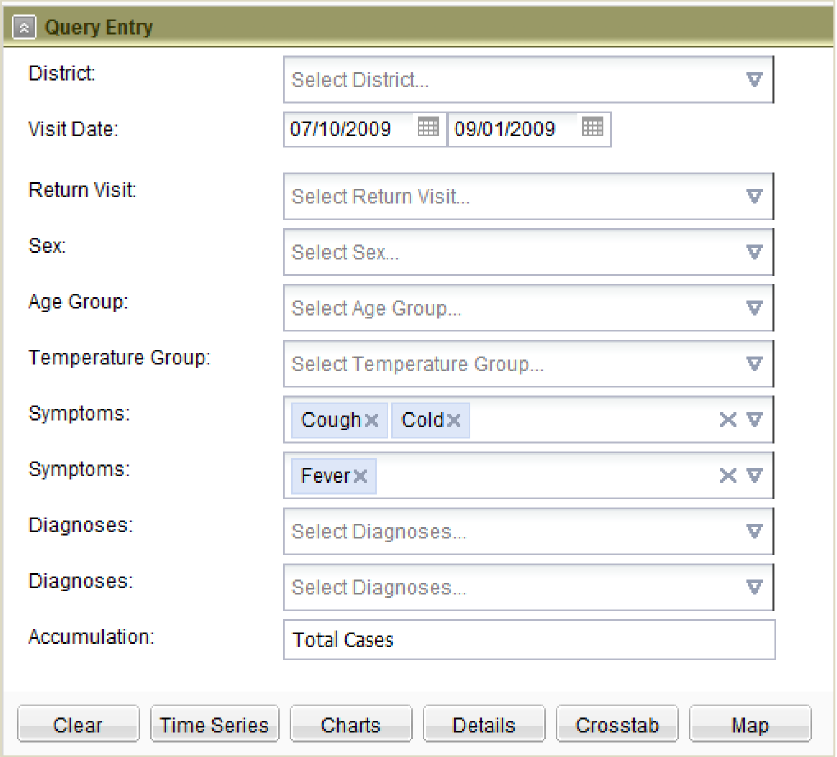


Report Date: 06/03/2009 09/01/2009

District: Select District...

Symptom Accumulations: Select Symptom Accumulations...

3. Select or type the values in the data fields. Most list fields allow for multiple selections. Within a single field, multiple selections are treated as a logical OR. Between fields, selections are treated as a logical AND. For example, the **Individual Patient Data** query form shown below will look for records that contain symptoms of Cough or Cold, and records that contain a symptom of Fever – operationally, (Cough OR Cold) AND (Fever).



District:	Select District...
Visit Date:	07/10/2009 09/01/2009
Return Visit:	Select Return Visit...
Sex:	Select Sex...
Age Group:	Select Age Group...
Temperature Group:	Select Temperature Group...
Symptoms:	Cough Cold
Symptoms:	Fever
Diagnoses:	Select Diagnoses...
Diagnoses:	Select Diagnoses...
Accumulation:	Total Cases

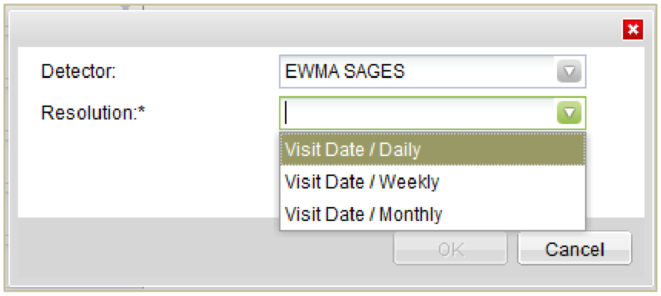
NOTE: You are not required to enter filter values on the query form. The resulting time series chart would be filtered only by the date range.

4. Click **Time Series**. A pop up window appears with Detector and Resolution options.
5. (Optional) Select a detection algorithm from the **Detector** list.

Detector refers to the mathematical algorithm used to identify statistically significant increases in temporal data, often called alerts. The default value for this variable is no detection, and the resulting time series chart will display only the counts. Alerts will be displayed when an algorithm is selected. The algorithms available in OpenESSENCE include SAGES_CUSUM (Cumulative Summation), SAGES_EWMA (Exponential Weighted Moving Average), SAGES_GS (Generalized Adaptive Smoothing), CDC-C1, CDC-C2, and CDC-C3. All anomaly detectors have pros and cons and their utility and validity are associated with the volume and distribution of data being examined. The SAGES detectors were developed by JHU/APL specifically for the types of environments in which SAGES operates. US Centers for Disease Control and Prevention (CDC) Early Aberration Reporting System algorithms (EARS C1/C2/C3) are also included as optional detection algorithms. More information about the SAGES detectors can be found in

Appendix B of this guide, including guidelines on selecting an appropriate detector based on the type of data being collected.

6. From the **Resolution** list, select **Visit Date / Daily**, **Visit Date / Weekly**, or **Visit Date / Monthly**. These resolutions will aggregate the data by day, week, or month respectively.



Detector: EWMA SAGES

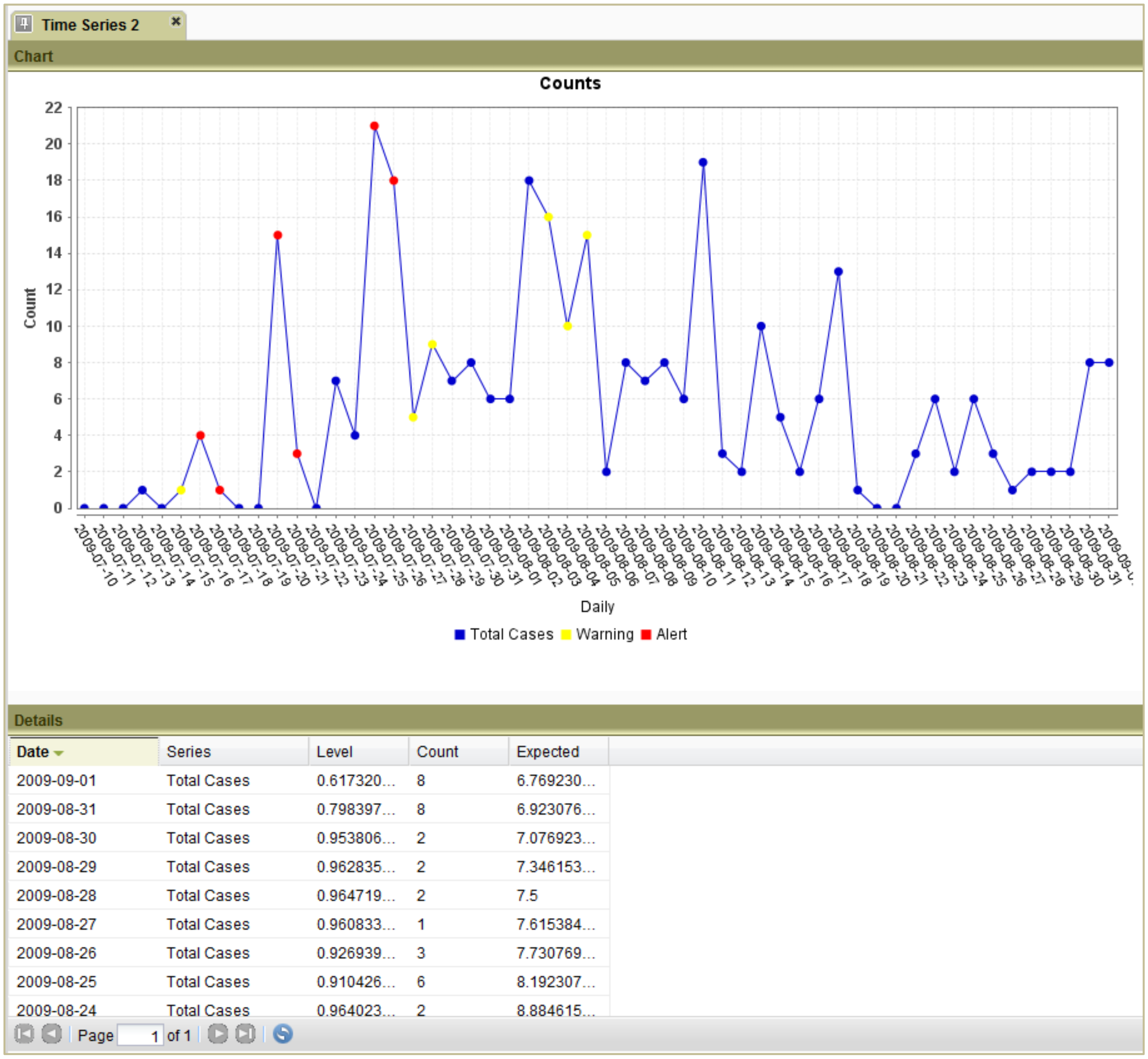
Resolution:*

Visit Date/Daily will display the time period specified in the query **Visit Date** field day by day.

Visit Date/Weekly will display the time period specified in the query **Visit Date** fields as epi-weeks. An epi-week starts on Sunday and ends the following Saturday. The first week in the Visit Date/Weekly display will include 7-day epi-week, even if the date you specify starts after the beginning of that week. The last week's total will only include dates from the start of the epi-week through the last date specified.

Visit Date/Monthly will display the time period specified in the query **Visit Date** fields as months. A month starts on the 1st of the month and ends on the last date of the month. The first month in the Visit Date/Monthly display will begin on the first day of the date entered into the query even if this date is later than the first of the month. The last month's total will only include dates from the 1st through the last date specified.

7. Click **OK.** A new tab appears displaying the time series chart. The time series tab is divided into two panels. The top panel displays the time series chart. The lower panel displays details, including detection results, for the time series. If a detection algorithm was not selected, the values in the **Count** column will equal the values in the **Expected** column.



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...

Page 1 of 1

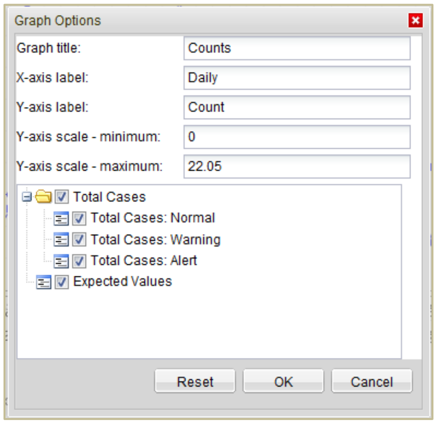
8. Use the toolbar buttons at the top right of the chart panel to resize the chart to fit the window; edit chart properties such as graph title, X/Y axis labels, and Y axis scale; or download the chart to a PNG file.

   Resize the chart to fit the window

   Edit chart properties such as graph title, X/Y axis labels, and Y axis scale

   Download the chart

9. If you have selected detection for the time series, you have the option to see the expected values on the graph as well as the actual values. To see expected values on the graph, click the **Edit Chart Properties** button, and check the **Expected Values** check box. Click **OK**.

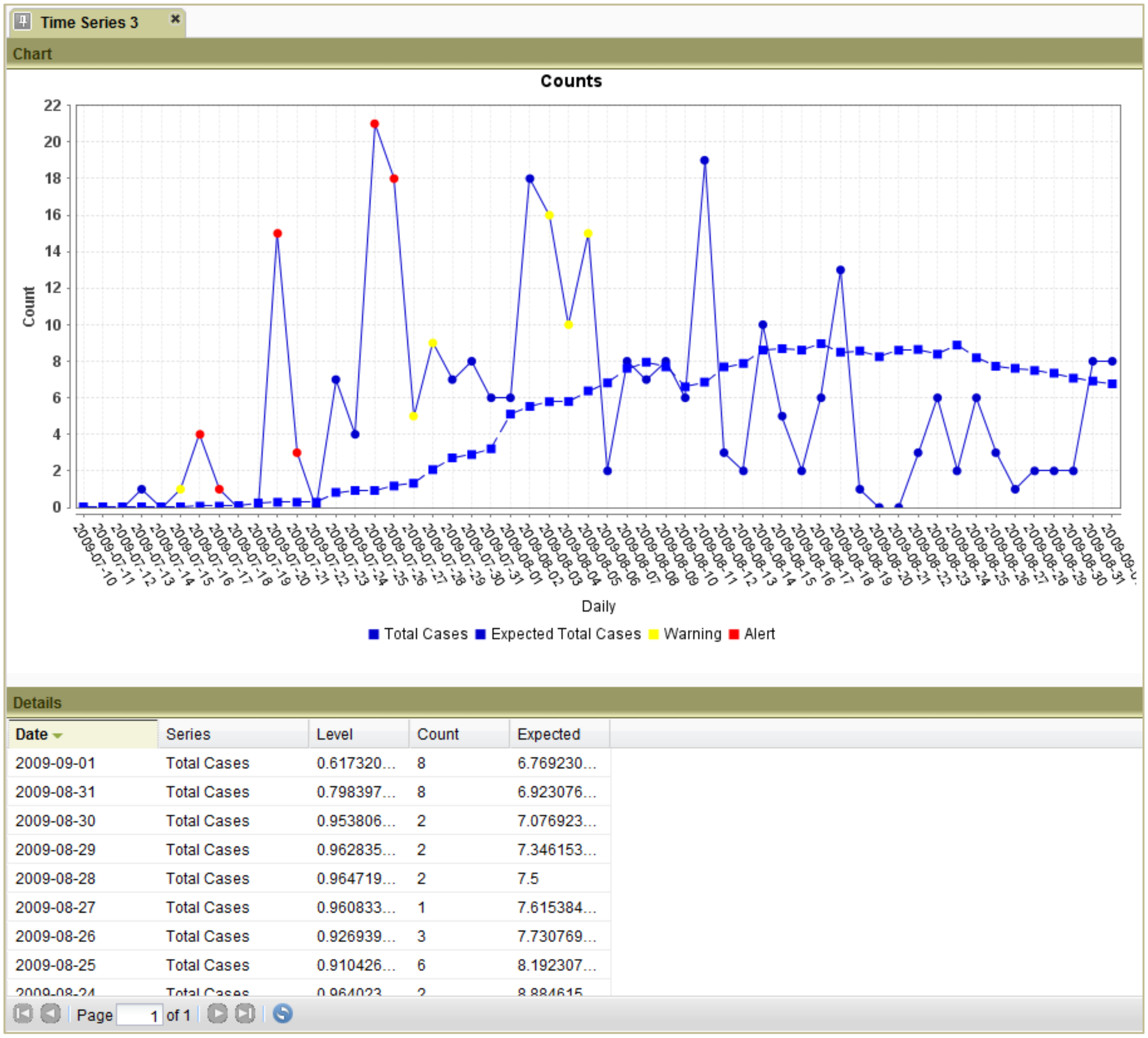


Graph Options

Graph title:	Counts
X-axis label:	Daily
Y-axis label:	Count
Y-axis scale - minimum:	0
Y-axis scale - maximum:	22.05

- ☑ Total Cases
  - ☑ Total Cases: Normal
  - ☑ Total Cases: Warning
  - ☑ Total Cases: Alert
- ☑ Expected Values

Expected values will be shown on a separate line on the graph.



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

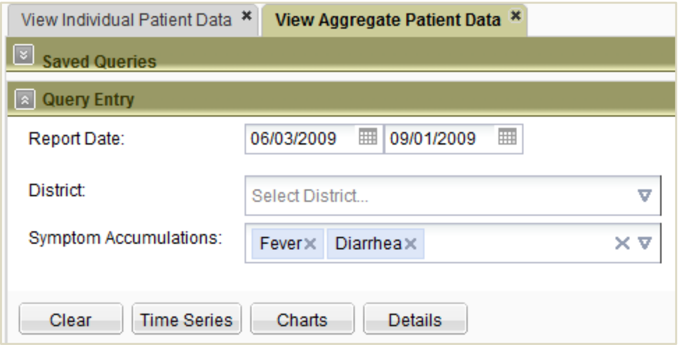
10. Click a point in the time series chart to view details for the selected group/resolution. The details will appear in a new tab. To return to the time series chart, click the **Time Series: Results** tab.

    Alerts are shown on the time series graph as yellow or red date markers (dots). An alert marks a day or time period when the detection algorithm indicates that the record count is statistically significantly greater than expected based on recent data. Alerts marked on the time series as red dots indicate that the increase is statistically significantly with a p-value of < 0.01, the yellow warning dots correspond to a p-value of < 0.05, and the blue dots indicate the value was within expected norms. When the cursor is placed on a dot, a pop-up window shows the date, the p-value level, the count, and the algorithm expected value of the point on which the cursor is sitting.

Create a Multi-Series Time Series Chart

The **View Aggregate Patient Data** option supports the creation of a time series chart containing multiple symptom accumulations. Here is an example:

1. From the **Navigation** menu, select **View Aggregate Patient Data**. A new tab appears containing the **View Aggregate Patient Data** query form.

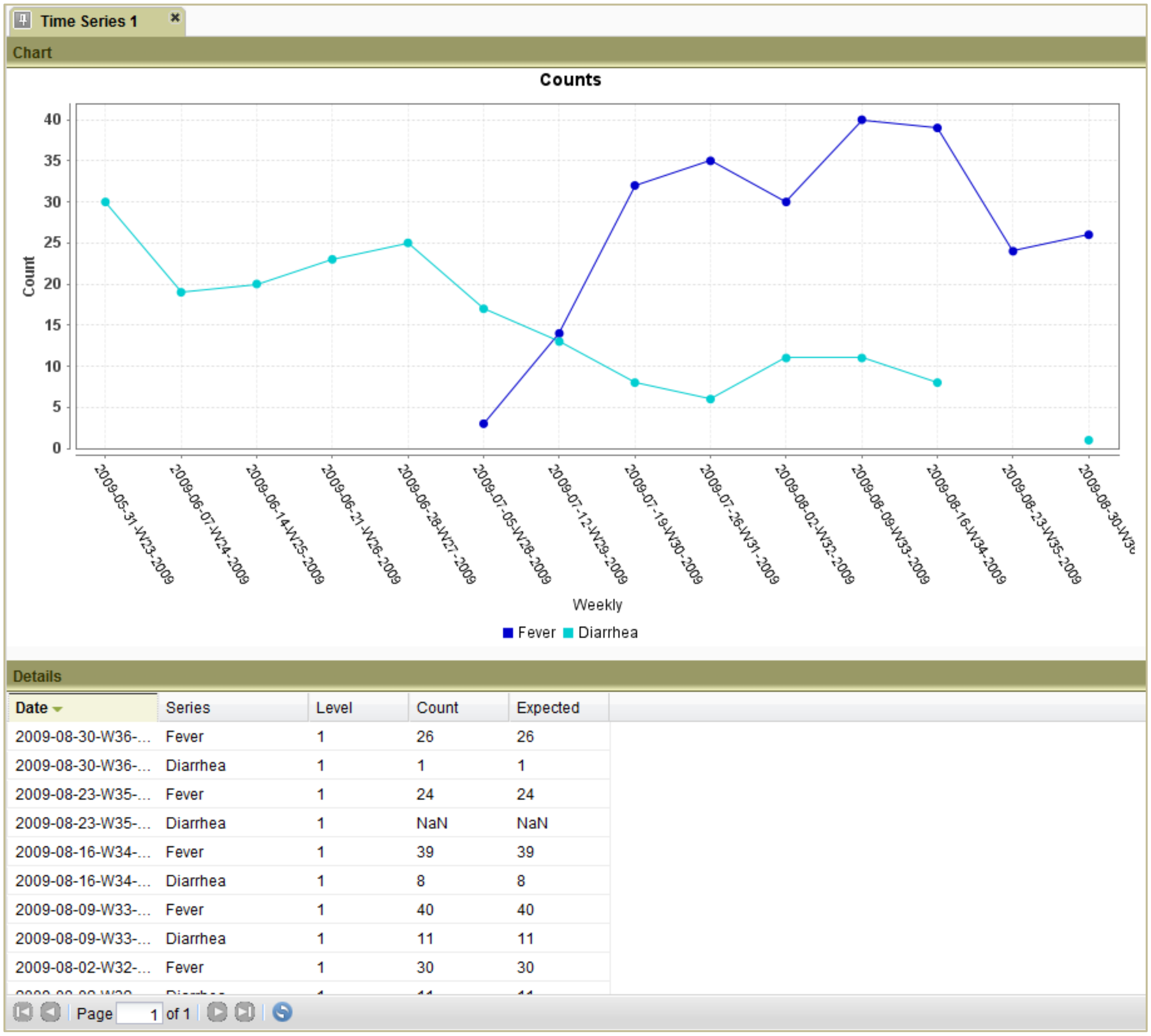


Report Date: 06/03/2009 09/01/2009

District: Select District...

Symptom Accumulations: Fever× Diarrhea×

2. Enter the desired date range in the **Report Date** field.
3. If desired, select one or more districts (geographic areas) to filter by. If no district is selected, then data from all districts will be returned.
4. In the **Symptom Accumulations** field, select the symptom accumulations you wish to see. In the example, we selected fever and diarrhea.
5. Click **Time Series**.
6. Select the **Detector** and **Resolution**. In the example, we selected no detection and weekly resolution. The weekly counts for fever and diarrhea appear as separate plotted lines on the chart. The details of each time series are listed by date and series in the lower panel.



Date	Series	Level	Count	Expected
2009-08-30-W36-...	Fever	1	26	26
2009-08-30-W36-...	Diarrhea	1	1	1
2009-08-23-W35-...	Fever	1	24	24
2009-08-23-W35-...	Diarrhea	1	NaN	NaN
2009-08-16-W34-...	Fever	1	39	39
2009-08-16-W34-...	Diarrhea	1	8	8
2009-08-09-W33-...	Fever	1	40	40
2009-08-09-W33-...	Diarrhea	1	11	11
2009-08-02-W32-...	Fever	1	30	30
2009-08-02-W32-...	Diarrhea	1	11	11

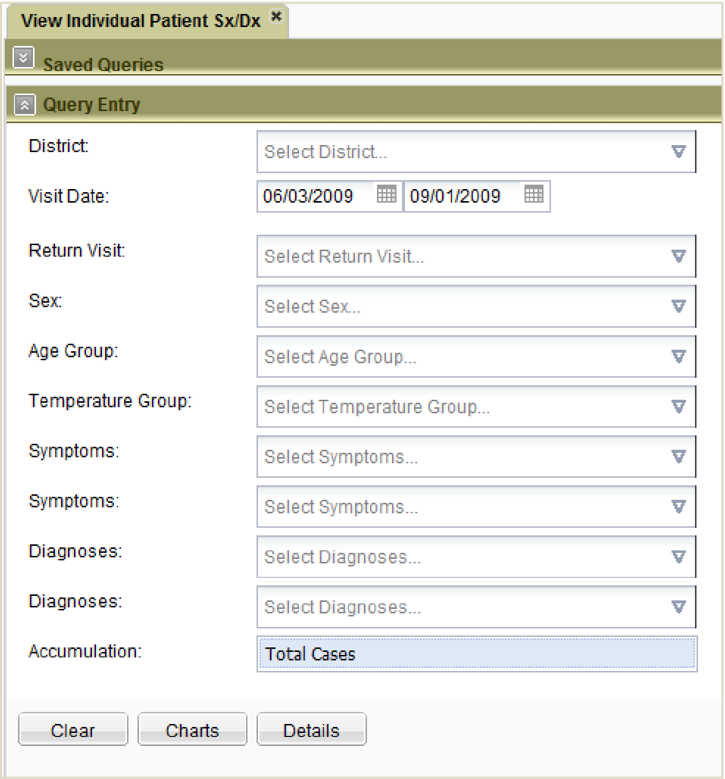
Create a Pie or Bar Chart

We will us the **View Individual Patient Sx/Dx** option to describe the pie/bar chart feature in OpenESSENCE. Multiple charts can be created at once by selecting multiple grouping and accumulation combinations. Charts can be downloaded in PNG (Portable Network Graphics) file format for use in other applications.

The **View Individual Patient Sx/Dx** allows you to view total cases by symptom and diagnosis based on individual patient data.

1. On the OpenESSENCE Demo screen, locate the **Navigation Menu** on the left.
2. Under **Analysis and Visualization**, select **View Individual Patient Sx/Dx**. A new tab appears in the right hand window of the Demo screen. The tab contains a query entry form, containing data fields which are used as filters.
3. Select or type the values in the data fields.

   NOTE: **Accumulation** is set to **Total Cases** by default and is not changeable.



District: Select District...

Visit Date: 06/03/2009 09/01/2009

Return Visit: Select Return Visit...

Sex: Select Sex...

Age Group: Select Age Group...

Temperature Group: Select Temperature Group...

Symptoms: Select Symptoms...

Symptoms: Select Symptoms...

Diagnoses: Select Diagnoses...

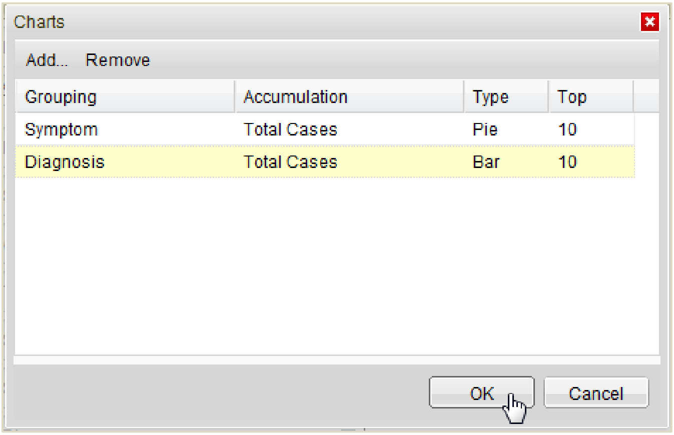
Diagnoses: Select Diagnoses...

Accumulation: Total Cases

4. Click **Charts**.

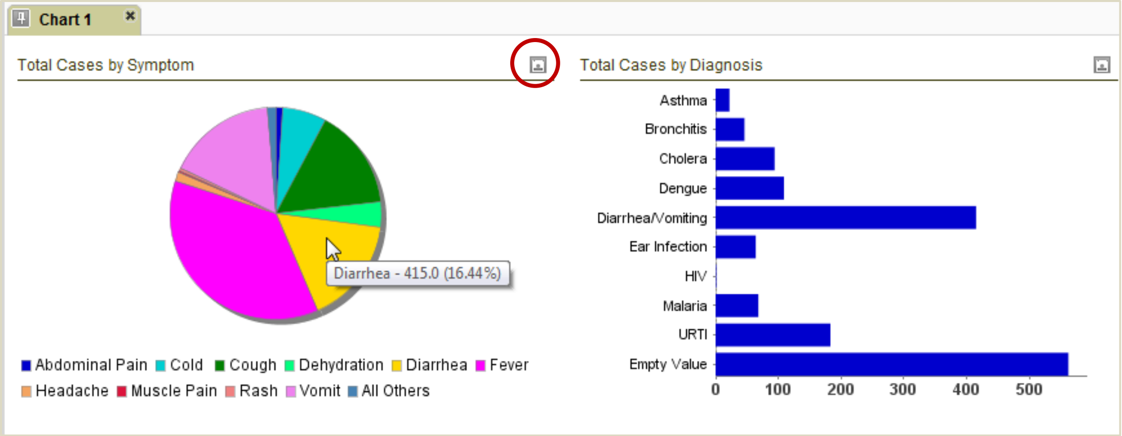
5. In the **Charts** dialog box, under **Grouping**, select the variable to be displayed in the chart. For our example, we will select **Symptom**.
6. **Accumulation** is fixed at **Total Cases**.
7. Under **Type**, select the type of chart to create (**Pie** or **Bar)**.
8. (Optional) Under **Top**, select the number of groupings (slices or bars) to show in the chart. In the example below, we have selected to show only the top 10 symptoms and diagnoses.

NOTE: You can create multiple charts at one time by clicking **Add** on the menu bar. Select **Grouping**, **Accumulation**, **Type**, and **Top** for each subsequent chart.



Grouping	Accumulation	Type	Top
Symptom	Total Cases	Pie	10
Diagnosis	Total Cases	Bar	10

The example charts below show the top 10 total cases by symptom in a pie chart and the top 10 total cases by diagnosis in a bar chart. Click the **Download Chart** icon at the top right of any chart to save a copy of that chart in PNG format. Position the cursor over any part of a chart to see count information about that section of the chart. Click a slice/bar in the chart to view details for the selected group/resolution. The details will appear in a new tab. To return to the chart, click the **Chart** tab.

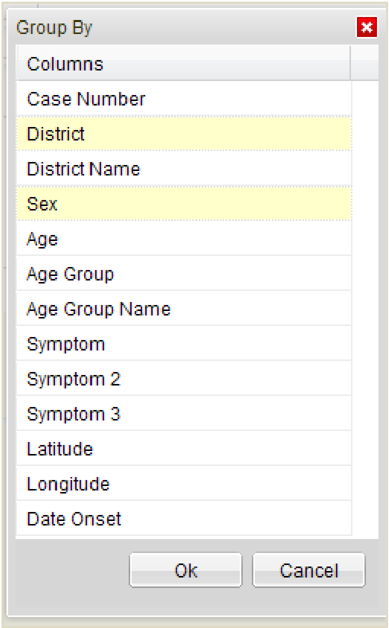


View Details

The Details feature displays records in a tabular grid format. The records can be aggregated, or grouped, by one or more columns. For example, if the **District** and **Sex** columns are selected, the **Details** grid will provide aggregation for each District and Sex combination. The grid can be exported to Microsoft Excel. We will use the Individual Patient Data option to illustrate the Details feature in OpenESSENCE.

1. From the **Navigation** menu, select **Individual Patient Data**. A new tab appears containing the **Individual Patient Data** query form.
2. If desired, enter values in the data fields to filter the data results.
3. Click **Details**.
4. Select the column(s) to group by.

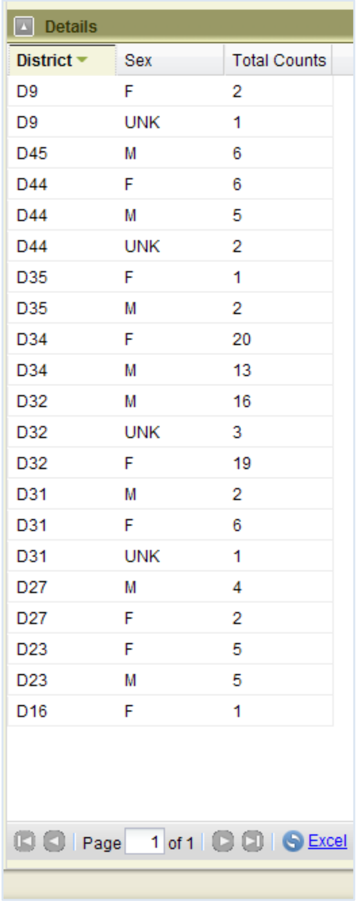
   NOTE: To select multiple columns individually, press and hold CTRL while selecting the columns. To select an entire range of columns, press and hold SHIFT while selecting the columns.



Group By

Columns
Case Number
District
District Name
Sex
Age
Age Group
Age Group Name
Symptom
Symptom 2
Symptom 3
Latitude
Longitude
Date Onset

5. Click **OK**. The grid results appear in a new tab named **Details**. Columns can be sorted by clicking on the column headings or reordered by dragging the column heading to the desired placement. Controls at the bottom of the grid enable searching through pages of data and export of the grid data to Microsoft Excel.



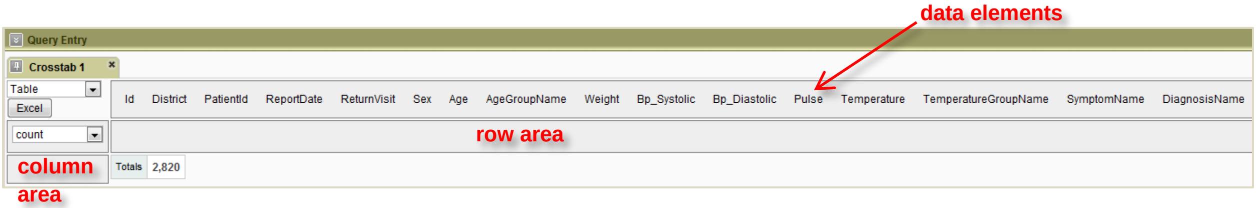
Details

District	Sex	Total Counts
D9	F	2
D9	UNK	1
D45	M	6
D44	F	6
D44	M	5
D44	UNK	2
D35	F	1
D35	M	2
D34	F	20
D34	M	13
D32	M	16
D32	UNK	3
D32	F	19
D31	M	2
D31	F	6
D31	UNK	1
D27	M	4
D27	F	2
D23	F	5
D23	M	5
D16	F	1

Create a Crosstab

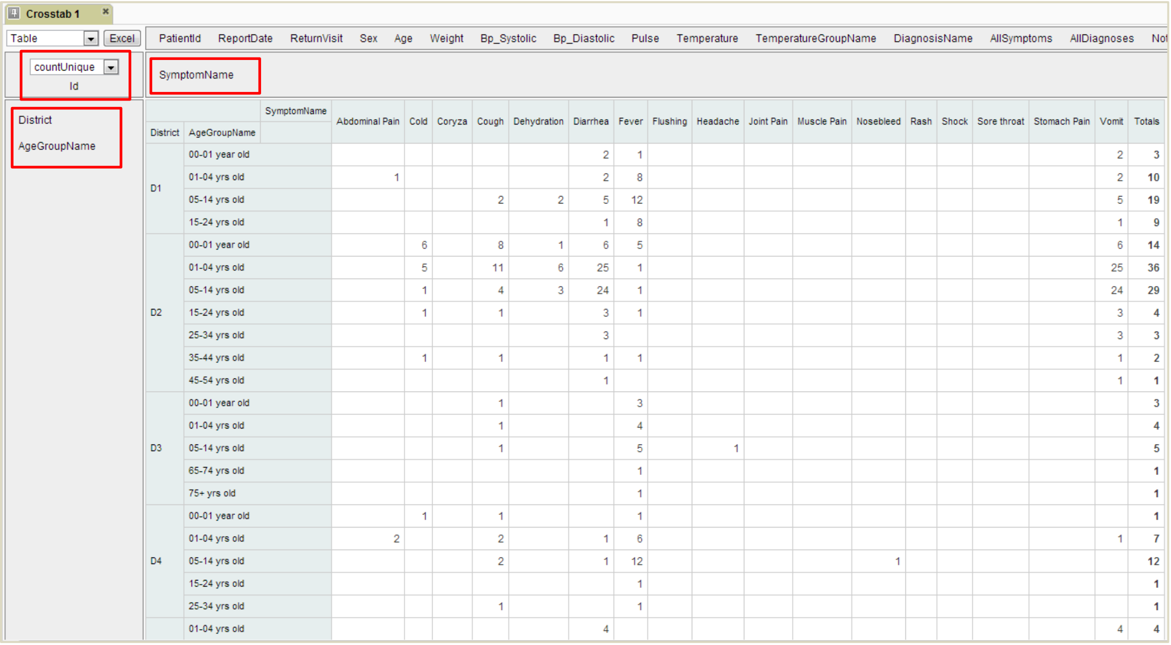
A crosstab report (or N x N x N table) provides the ability to display the frequency of data based on three variables, such as district, age group, and symptom. These reports use the concept of "pivot" tables to quickly create multi-dimensioned reports. We will use the Individual Patient Sx/Dx option to illustrate the Crosstab feature in OpenESSENCE.

1. From the **Navigation** menu, select **Individual Patient Sx/Dx**. A new tab appears containing the **Individual Patient Sx/Dx** query form.
2. If desired, enter values in the data fields to filter the data results.
3. Click **Crosstab**. The crosstab creation screen appears. From here, you can drag data elements to either the column or row area to create the crosstab report.



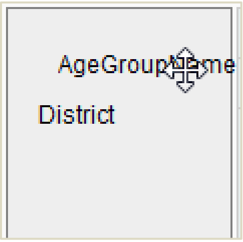
4. For our example, drag **District** to the column area, drag **AgeGroupName** to the column area, and drag **SymptomName** to the row area.
5. To make sure that we are counting unique records, select **countUnique** from the pull down list as shown below. Then drag **Id** to the area directly underneath the pull down list. This indicates that we want to count unique records (the **Id** variable represents unique records). The crosstab results will appear as shown.

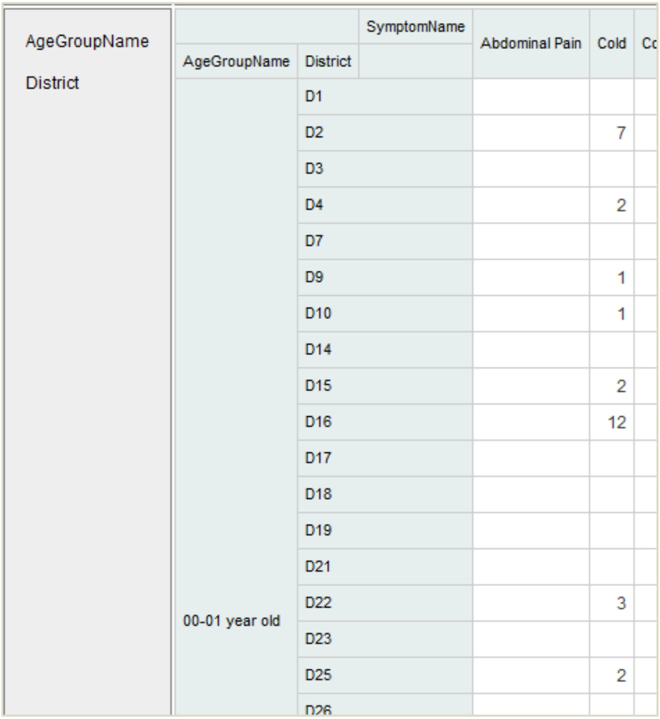
   NOTE: You should always use the **countUnique** option with the **Id** variable when generating a crosstab with Individual Patient data. If using aggregate data, you should select the **intSum** option from the pull down list, and drag the **TotalCases** variable underneath it.



	SymptomName	Abdominal Pain	Cold	Coryza	Cough	Dehydration	Diarrhea	Fever	Flushing	Headache	Joint Pain	Muscle Pain	Nosebleed	Rash	Shock	Sore throat	Stomach Pain	Vomit	Totals
District	AgeGroupName																		
D1	00-01 year old						2	1										2	3
	01-04 yrs old	1					2	8										2	10
	05-14 yrs old				2	2	5	12										5	19
	15-24 yrs old						1	8										1	9
D2	00-01 year old		6		8	1	6	5										6	14
	01-04 yrs old		5		11	6	25	1										25	36
	05-14 yrs old		1		4	3	24	1										24	29
	15-24 yrs old		1		1		3	1										3	4
	25-34 yrs old						3											3	3
	35-44 yrs old		1		1		1	1										1	2
	45-54 yrs old						1											1	1
D3	00-01 year old				1			3											3
	01-04 yrs old				1			4											4
	05-14 yrs old				1			5		1									5
	65-74 yrs old							1											1
	75+ yrs old							1											1
D4	00-01 year old		1		1			1											1
	01-04 yrs old	2			2		1	6										1	7
	05-14 yrs old				2		1	12					1						12
	15-24 yrs old							1											1
	25-34 yrs old				1			1											1
	01-04 yrs old						4											4	4

6. You can change the layout of the crosstab dynamically. For instance, the crosstab currently shows symptom counts by age group by district. If you wanted to show counts by district by age group, you would switch the positions of the **AgeGroupName** and **District** labels in the row area, by clicking and dragging **AgeGroupName** above **District**. (Notice that the cursor changes to a 4-sided arrow as you are dragging.)



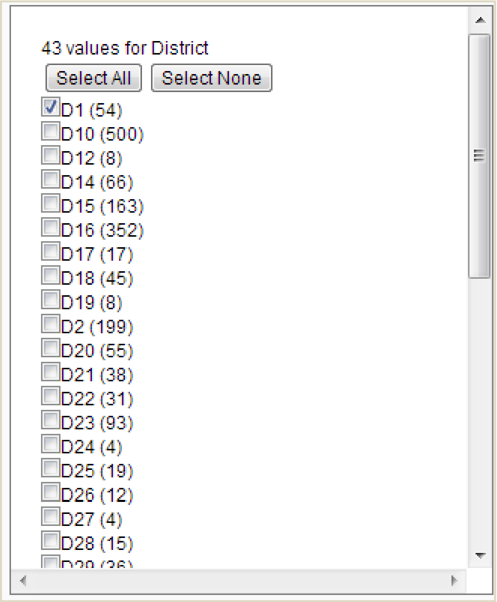


AgeGroupName

District

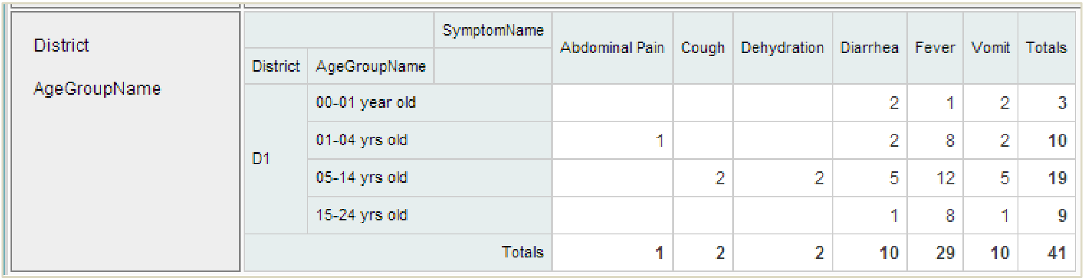
	SymptomName	Abdominal Pain	Cold	Co
AgeGroupName	District			
00-01 year old	D1			
	D2		7	
	D3			
	D4		2	
	D7			
	D9		1	
	D10		1	
	D14			
	D15		2	
	D16		12	
	D17			
	D18			
	D19			
	D21			
	D22		3	
	D23			
	D25		2	
	D26			

7. Recall that you can apply filters to the dataset prior to creating the crosstab (in step 2). You can also apply filters after the crosstab has been created. For example, suppose you want the crosstab to show only District 1 symptom counts. Double-click **District** in the column area. From the list of districts, select **D1**. (You can use the **Select All** and **Select None** options as needed). Now click anywhere on the crosstab to see the filtered results.



43 values for District

☑D1 (54)
☐D10 (500)
☐D12 (8)
☐D14 (66)
☐D15 (163)
☐D16 (352)
☐D17 (17)
☐D18 (45)
☐D19 (8)
☐D2 (199)
☐D20 (55)
☐D21 (38)
☐D22 (31)
☐D23 (93)
☐D24 (4)
☐D25 (19)
☐D26 (12)
☐D27 (4)
☐D28 (15)
☐D29 (36)



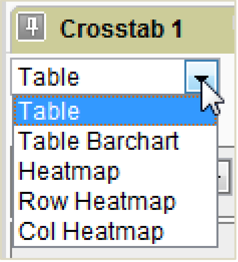
District

AgeGroupName

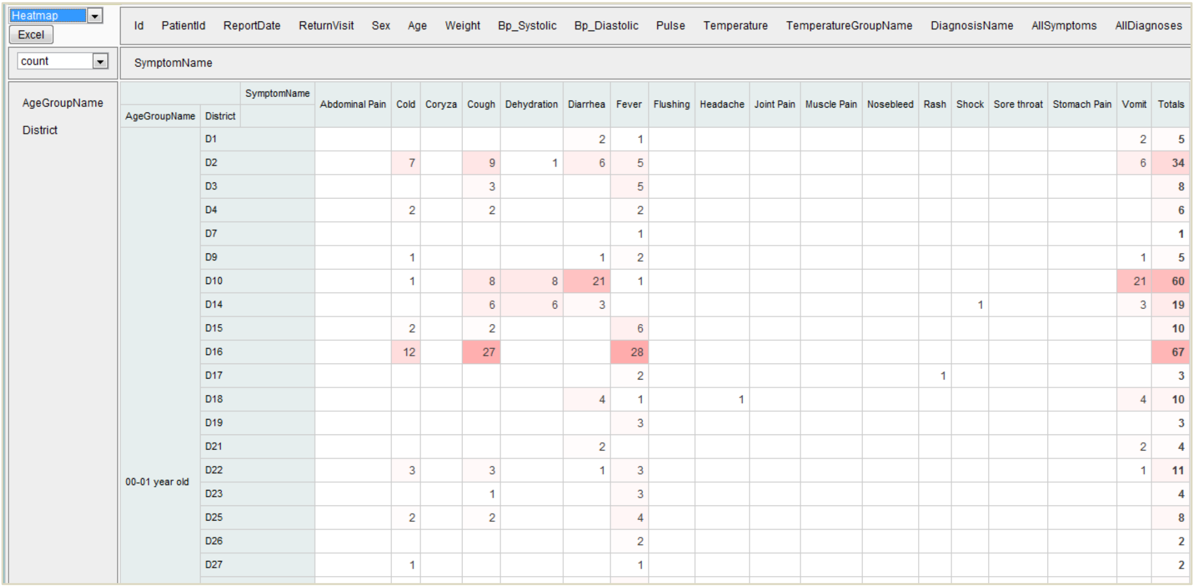
		SymptomName	Abdominal Pain	Cough	Dehydration	Diarrhea	Fever	Vomit	Totals
District	AgeGroupName								
D1	00-01 year old					2	1	2	3
	01-04 yrs old		1			2	8	2	10
	05-14 yrs old			2	2	5	12	5	19
	15-24 yrs old					1	8	1	9
		Totals	1	2	2	10	29	10	41

NOTE: You can use the same technique to apply filters to the other elements of the crosstab (**AgeGroupName** and **SymptomName**).

8. By default, the crosstab is a basic table. There are additional visualization features that can be applied to the crosstab. Click the pull down list in the upper left of the crosstab screen to apply additional visualizations to the table.



Examples of **Heatmap** and **Table Barchart** are shown below. The **Heatmap** uses color gradients to emphasize higher count values (the darker the color, the higher the value). The **Table Barchart** uses bars (as in a traditional bar graph) to emphasize variations in count value.



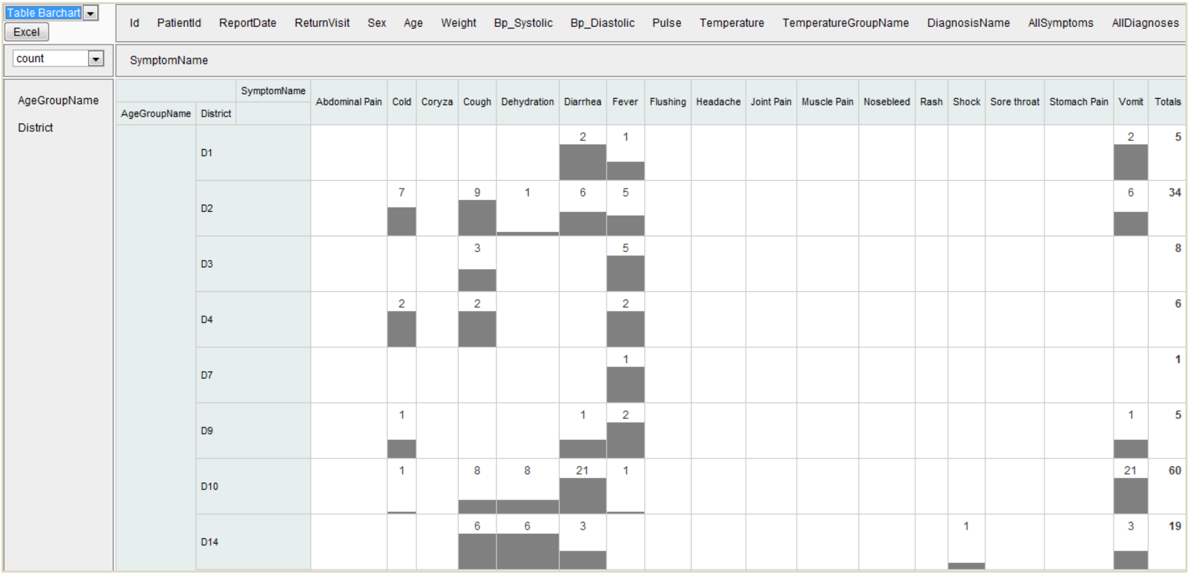
Id PatientId ReportDate ReturnVisit Sex Age Weight Bp_Systolic Bp_Diastolic Pulse Temperature TemperatureGroupName DiagnosisName AllSymptoms AllDiagnoses

SymptomName

AgeGroupName

District

	SymptomName	Abdominal Pain	Cold	Coryza	Cough	Dehydration	Diarrhea	Fever	Flushing	Headache	Joint Pain	Muscle Pain	Nosebleed	Rash	Shock	Sore throat	Stomach Pain	Vomit	Totals
AgeGroupName	District																		
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



Id PatientId ReportDate ReturnVisit Sex Age Weight Bp_Systolic Bp_Diastolic Pulse Temperature TemperatureGroupName DiagnosisName AllSymptoms AllDiagnoses

SymptomName

AgeGroupName

District

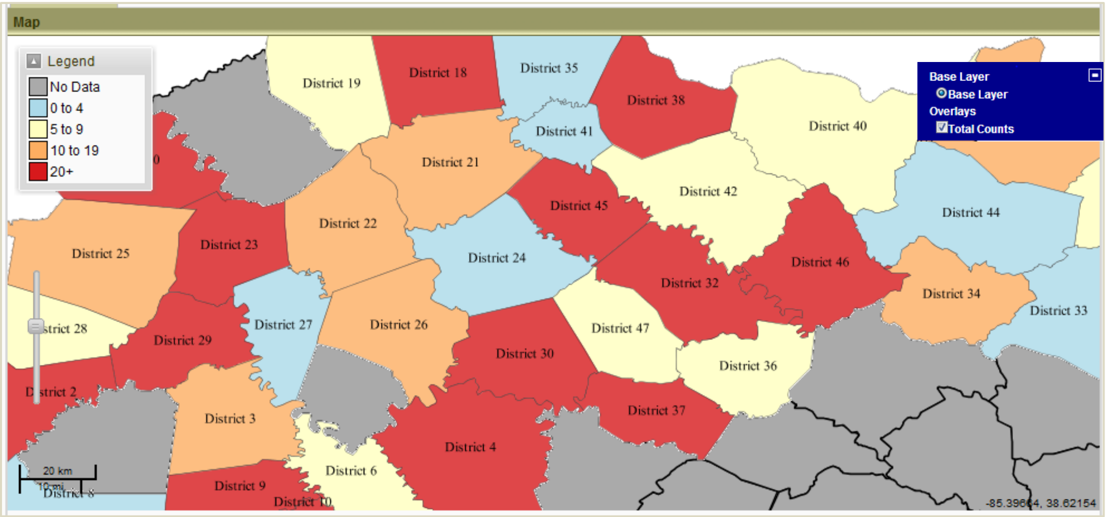
	SymptomName	Abdominal Pain	Cold	Coryza	Cough	Dehydration	Diarrhea	Fever	Flushing	Headache	Joint Pain	Muscle Pain	Nosebleed	Rash	Shock	Sore throat	Stomach Pain	Vomit	Totals
AgeGroupName	District																		
	D1						2	1										2	5
	D2		7		9	1	6	5										6	34
	D3				3			5											8
	D4		2		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

9. You can also export the data from the crosstab to Microsoft Excel using the **Excel** option at the upper left.

Generate a Map

The Map feature is only available in the **Individual Patient Data** option. A region map is created by clicking **Map** from the query form.

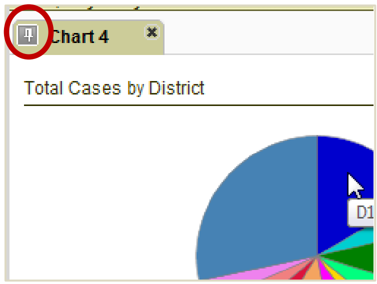
1. From the **Navigation** menu, select **View Individual Patient Data**. A new tab appears containing the **Individual Patient Data** query form.
2. If desired, enter values in the data fields to filter the data results.
3. Click **Map**. A geographic map appears in a new tab. The Legend at the top left of the map describes the meaning of the map colors in relation to case count. The plus sign on the right side of the map controls which layers are displayed. In this example, a base layer displays the district geographies as gray polygons without district name labels. And an overlay layer displays the Total Counts for the district geographies as non-gray shaded polygons with district name labels. Only districts with data represented in the query results are non-gray shaded and labeled.
4. To zoom in or out, adjust the zoom level using the control on the left.
5. Click and drag in any direction to move the map as needed.



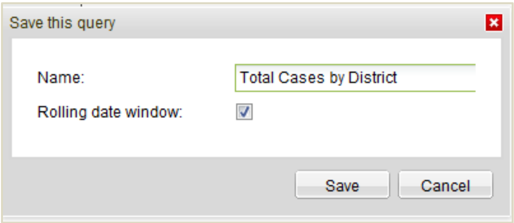
Save a Query

If there is a query that you will be using frequently, you can save the query so that it can be run again without having to enter the filters again. For example, you may want to see the same time series chart every day. Rather than creating the query every day, you can create it once, and save it to be used again. To save a query:

1. Click the pin icon from the query results tab of the query you wish to save. The query results tab is the tab in which your time series charts, pie/bar charts, maps, or detail grids appear.



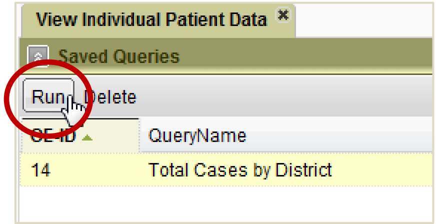
2. Enter a name for the saved query and select whether or not you want to use the rolling date option. If the rolling date option is selected, then the specific start and end dates of the query are ignored. Instead, only the query's length of time is preserved. The end date will be the date that the saved query is run. If the rolling date option is not selected, then the query will use the same dates specified in the query. The saved query in the following example is saved with the name Total Cases by District.


Save this query

Name: Total Cases by District

Rolling date window: ☑

3. To run a saved query, expand the **Saved Queries** panel, select the query, and click **Run** (or double-click the query).



QE ID	QueryName
14	Total Cases by District

ADMINISTRATIVE REPORTS

The OpenESSENCE demo site contains two examples of administrative reports that focus on tracking the number of reports per geographic site as well as the time between patient visit date and report date. These types of administrative reports can be useful for determining the timeliness of data being entered in OpenESSENCE.

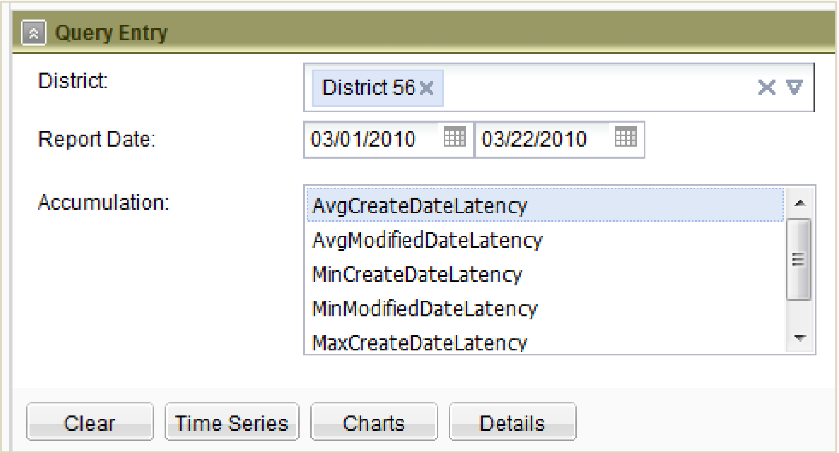
Aggregate Data Latency Report

For each data record in OpenESSENCE, the following date values exist:

- Visit Date: User defined date indicating the date of the patient visit (or date of the aggregate daily report)
- Create Date: System defined date indicating the date the record was created in the database
- Modified Date: System defined date indicating the date the record was last modified

The Aggregate Data Latency Report analyzes the time difference (in days) between the Create Date and the Visit Date (referred to as Create Date Latency) which directly indicates the timeliness of data entry; and the difference between the Modified Date and Create Date (referred to as Modified Date Latency).

1. From the **Navigation** menu, select **View Aggregate Data Latency**.
2. In the query form, enter the geographic district, date range, and type of latency you want to see.



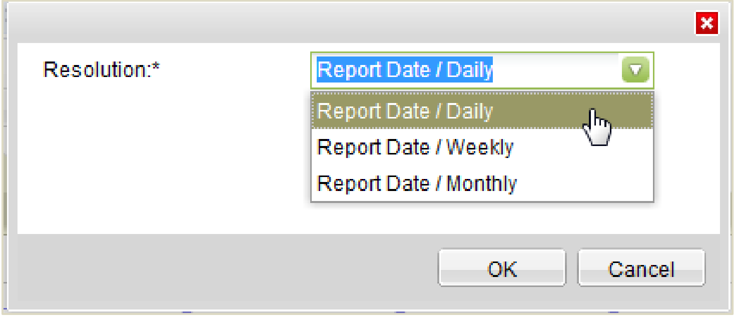
Query Entry

District: District 56

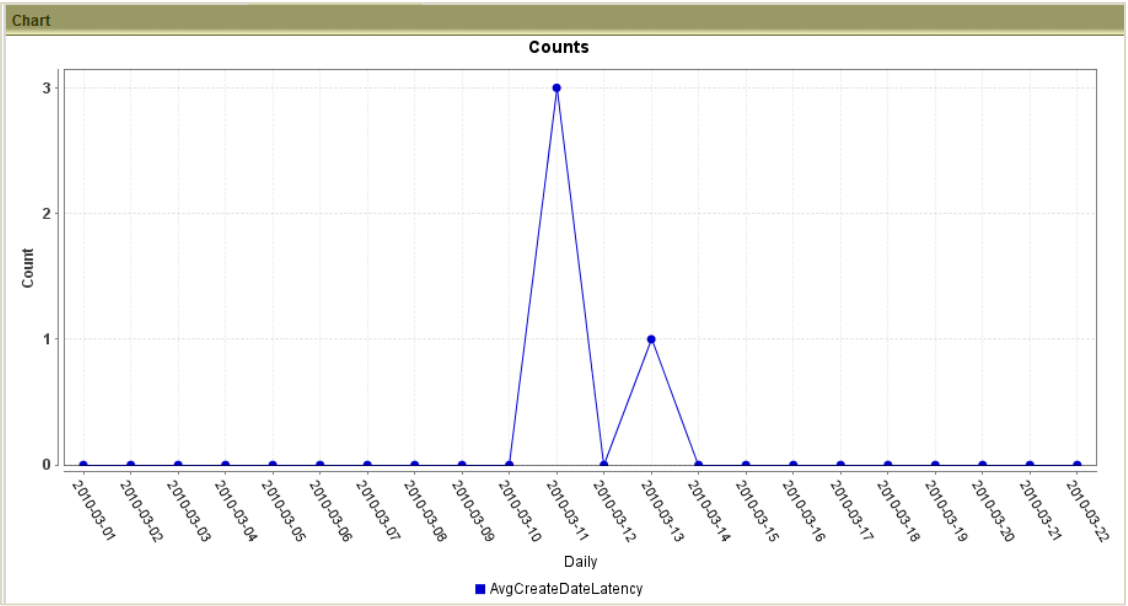
Report Date: 03/01/2010 03/22/2010

Accumulation: AvgCreateDateLatency

3. Click **Time Series**.
4. Select the **Resolution** (for our example, choose **Daily**).


Resolution:* Report Date / Daily

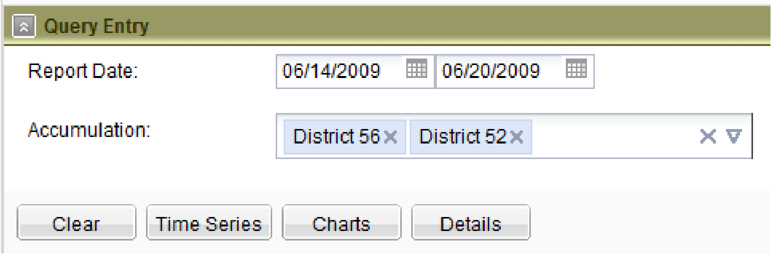
5. Click **OK**. The Data Latency time series appears on a new tab. The peaks indicate days when there was a lag (latency) between the Visit Date and the Create Date for District 56 for the given date range.



Aggregate Site Report

The Aggregate Site Report counts the number of aggregate reports created by a given district(s) on a daily, weekly, or monthly basis. For example, if a district is expected to report aggregate data once a day, then the daily count should be one. A daily count in the aggregate site report is zero indicates that the district did not enter the daily aggregate report.

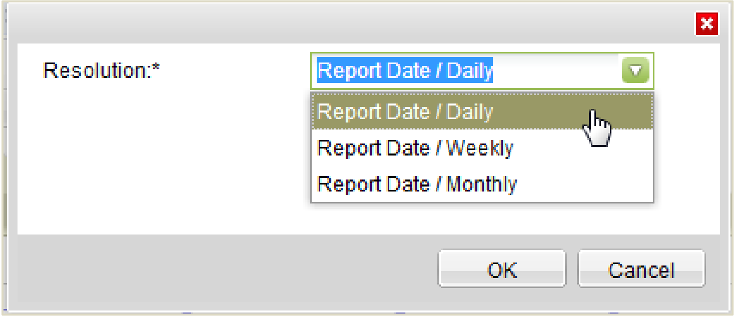
1. From the **Navigation** menu, select **View Aggregate Site Report**.
2. Select the **Report Date** range.
3. In the **Accumulation** field, select the district(s) for which you want to view the report counts.



Report Date: 06/14/2009 06/20/2009

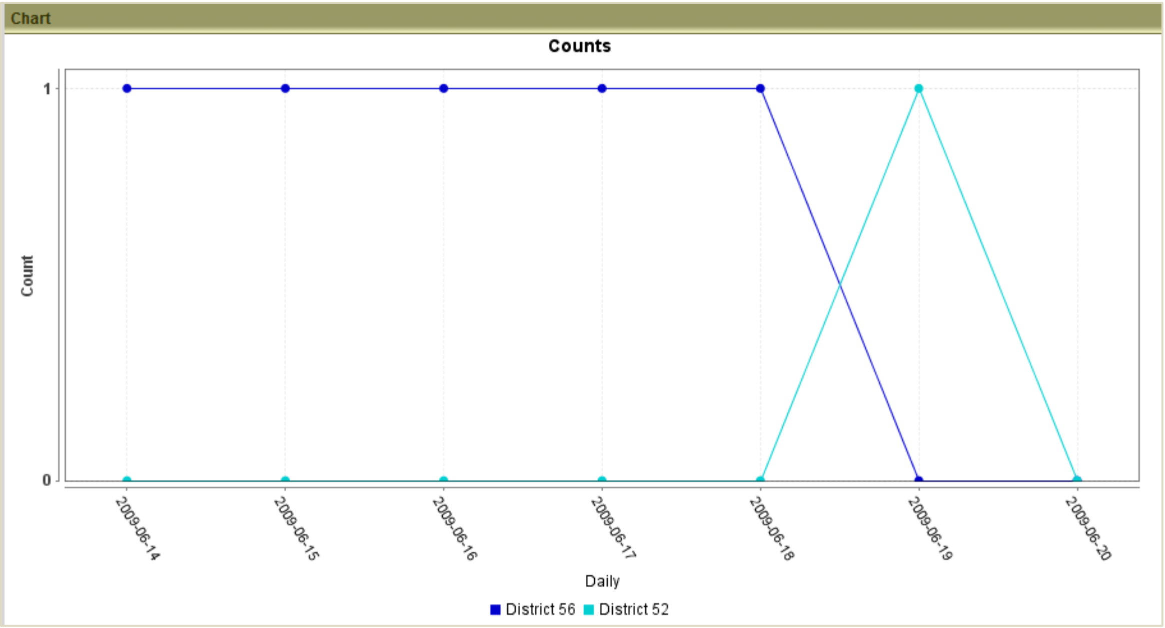
Accumulation: District 56 × District 52 ×

4. Click **Time Series**.
5. Select the **Resolution** (for our example, choose **Daily**).



Resolution:* Report Date / Daily

6. Click **OK**. The Aggregate Site Report appears in a new tab.



Appendix A: Description of Demo Data Set

The population represented in the demo data is loosely modeled after military personnel and their dependents seen in field clinics in a resource poor area. The demo data is fabricated data; some is modeled after real data and some is completely imaginary. The user may, therefore, encounter some biological inconsistencies in the data. These are generally irrelevant to the purpose of the data set, which is to showcase the functionality of OpenESSENCE.

Since the population represents a group of young adults and their dependents, the age distribution is skewed left, with more than 75% under 15 years of age. Somewhat more than 50% are male (52% vs. 47%). Gastrointestinal and fever related illnesses predominate. Looking at all illnesses during the entire time spanned by data collection, there are marked increases in reported illnesses in April 2009 and July/August 2009. More subtle increases occur in early June, December and February/March. The increases in April are mostly due to a continuing gastrointestinal outbreak that spread through the community. In contrast the increases in July/August and February 2010 are fever-related illnesses, primarily dengue, but also malaria and some illnesses diagnosed just as ‘Fever’.

Table 1: Number of Reports by Age Group

Age Group	Number of reports (%)
00-01 year old	431 (12%)
01-04 yrs old	1335 (36%)
05-14 yrs old	1350 (37%)
15-24 yrs old	238 (6%)
25-34 yrs old	120 (3%)
35-44 yrs old	87 (2%)
45-54 yrs old	59 (2%)
55-64 yrs old	32 (1%)
65-74 yrs old	10 (0%)
75+ yrs old	36 (1%)
Total	**3698 (100%)**

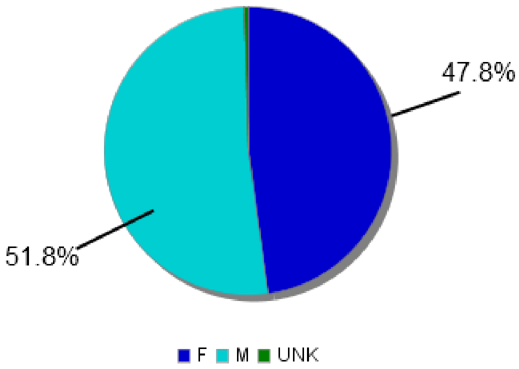


Figure 5: Reports Breakdown by Sex

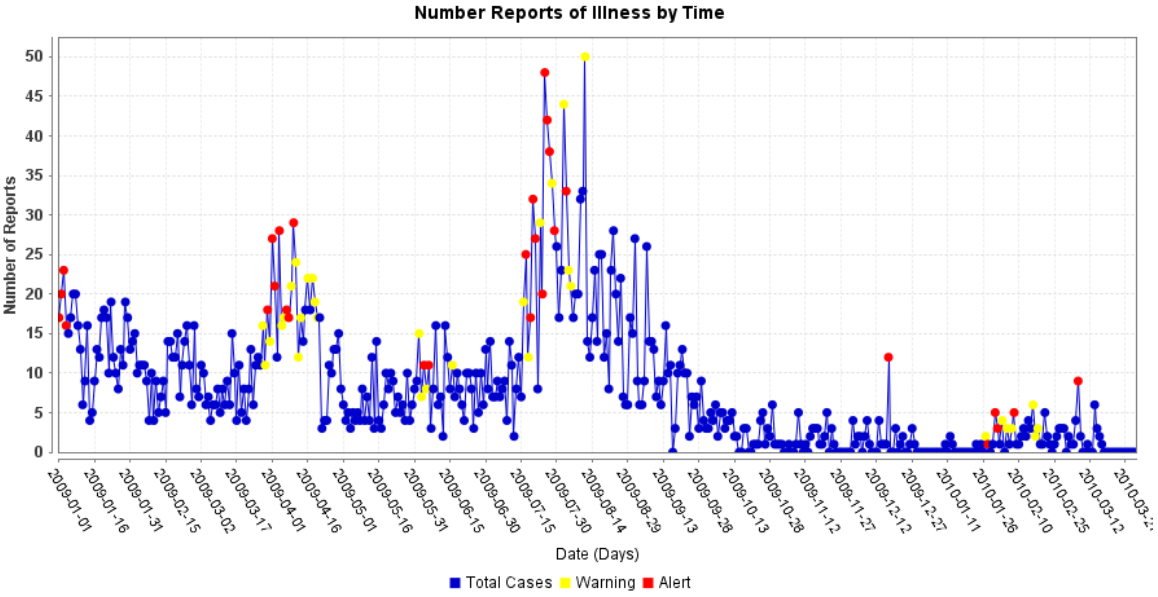


Figure 6: Number of Reports of Illness by Time

Appendix B: SAGES Detection Algorithms

SAGES_EWMA (Exponential Weighted Moving Average)

The adaptation of the Exponential Weighted Moving Average (EWMA) chart was developed using a large group of time series representative of SAGES syndromic data. The algorithm calculates a weighted average of recent time series values. The weighting is according to how recent each value is with the greatest weight on the most current observation and the weight decreases exponentially with the age of the observation. For the test statistic, an updated baseline mean value is subtracted from this weighted average, and the difference is divided by a standard error estimate to get a test value that is comparable for large and small data scales. To accommodate sparser time series, the standard error estimate is bounded from below to avoid excessive test values. For statistical significance, p-values are obtained with a standard t-test using the baseline length minus one for degrees of freedom. SAGES_EWMA was selected as the most effective method for monitoring daily time series based on its performance on nearly 100 documented outbreaks from multiple geographic regions. It is most effectively applied to time series without substantial day-of-week effects or other systematic behavior.

SAGES_CUSUM (Cumulative Summation)

The adaptation of the CUSUM chart was also developed from time series representative of SAGES syndromic data. SAGES_CUSUM maintains a cumulative summation, a running sum of differences from the baseline mean. This algorithm was selected as the most effective method for monitoring weekly time series, again based on performance with data containing known historical events. Like the SAGES_EWMA, it is most effectively applied to time series without substantial day-of-week effects or other systematic behavior. However, the approach is qualitatively different. This implementation is a one-sided test that monitors for excessive increases but not decreases; as such, alerts will be triggered only for increases and not for decreases. For values below the mean, this sum is not permitted to go negative but will stay at zero until the mean is exceeded. For values above the mean, each difference above *k* standard deviations is scaled by dividing by the current baseline standard deviation, and the cumulative summation is the sum of these scaled differences. For statistical significance, p-values are obtained from lookups from a table derived from a stratified collection of syndromic time series.

Like the EWMA_SAGES method, the CUSUM implementation for SAGES bounds the standard error from below. The standard error from the baseline for each day is used to divide and thus normalize successive differences. Without this lower bound, if the data are sparse and the entire baseline interval has very few counts, the standard error will be near zero, and extremely high values of the statistic will result and be carried forward. This safeguard makes both the EWMA and CUSUM methods applicable for monitoring sparse data. However, the CUSUM is slightly preferable for sparse data because the EWMA weighting of differences is based strictly on the age of the observation. If data are sparse, excessive counts due to an event of concern may be separated by several days, so that alerting by the CUSUM approach is more likely.

SAGES_GS (Generalized Adaptive Smoothing)

This algorithm applies generalized adaptive smoothing to treat time series that may exhibit sparse behavior or larger-scale periodic behavior. For day-of-week and trend effects, it captures localized behaviors that are not suitable for a fixed-effects regression approach. Generalized smoothing was chosen because it does not assume changes on a given day-of-week but infers these from the data, and it can accommodate regular data gaps, as in weekend closings. When time series counts are sparse, it converts to a basic weighted smoothing technique as in SAGES_EWMA. For time series known to have systematic effects that may cause bias, SAGES_GS is recommended. Otherwise, the simpler SAGES_EWMA or SAGES_CUSUM should be used.

CDC Early Aberration Reporting System C1, C2, and C3 Algorithms

The purpose of these algorithms is to detect general data aberrations. Algorithms C1, C2, and C3 of the Early Aberration Reporting System (EARS) developed at the Centers for Disease Control and Prevention are used in many U.S. states and in numerous foreign countries. They are included in OpenESSENCE because of their wide application. The purpose of these algorithms is to detect general data aberrations. They are categorized as adaptive control chart algorithms.

The C1 algorithm subtracts the daily count from the mean of a moving baseline ending the previous day. In effect, it then divides this difference by the standard deviation of counts in that baseline. If the result exceeds 3, indicating an increase above the mean of more than 3 standard deviations, an alert is issued.

The C2 algorithm does the same calculation but imposes a 2-day buffer between the test day and the baseline.

The C3 algorithm is a more sensitive version of C2 that adds the values from the 2 previous days if they do not exceed the threshold. All three algorithms use the same criterion of an increase of at least 3 baseline standard deviations above the sliding baseline mean.

An important implementation detail is that OpenESSENCE does not use the standard 7-day baseline because substantial experience has shown that for many time series, such a short baseline gives an unstable statistic that can lead to a loss of confidence in the results. The implemented baseline is 28 days. There are no other changes to the standard EARS methods, including retention of the flat 3-standard-deviation threshold regardless of the data stream.

These algorithms take no account of systematic data behavior such as day-of-week effects or seasonal trends. C3 is the only one of these methods with sensitivity to gradual outbreak effects, but it is known to produce high alarm rates. For all three methods, threshold data values for alerting may fluctuate noticeably from day to day.

Selecting a Detector

The following table provides a guideline for selecting a detector:

Legend:
★ = preferred
⯪ = acceptable
☆ = not recommended

Criteria	Daily data	Weekly, monthly data	>= 8 weeks baseline data	< 8 weeks baseline data	Sparse data	Strong day of week effect	Lots of missing data	Strong seasonal effect
SAGES_ EWMA	⯪	★	★	★	⯪	☆	⯪	⯪
SAGES_ CUSUM	⯪	★	★	★	★	☆	★	⯪
SAGES_ GS	★	⯪	★	⯪	⯪	★	⯪	★
EARS C1/C2/ C3	⯪	⯪	⯪	⯪	☆	☆	☆	⯪

NOTES:

1. SAGES_GS requires a minimum of 2 weeks of training data to run.
2. At least 4 weeks of training data are recommended for all detectors.
3. "Sparse" data are understood as series where the majority of values are usually zero.

Only SAGES_GS handles unusual or evolving day-of-week patterns.