Analyzing social media and structured data with InfoSphere BigInsights

Get a quick start with BigSheets

Cynthia M. Saracco (saracco@us.ibm.com)
Senior Software Engineer
IBM

Anshul Dawra (adawra@us.ibm.com)
Senior Software Engineer
IBM

Skill Level: Introductory

Date: 07 Jun 2012

If you want to work with "big data" without writing code or scripts, you'll want to look into BigSheets. BigSheets is a spreadsheet-style tool for business analysts provided with IBM® InfoSphere® BigInsights, a platform based on the open source Apache Hadoop project. This article teaches you the basics of using BigSheets to analyze social media and structured data collected through sample applications provided with BigInsights. You'll learn how to model this data in BigSheets, manipulate this data using built-in macros and functions, create charts to visualize your work, and export the results of your analysis in one of several popular output formats.

You've probably heard about big data and its impact on business analysis. And maybe you're wondering what insights your organization might glean from capturing, processing, and managing big data collected from websites, electronic sensors, or software logs, along with traditional data you already have. Certainly, there's no shortage of open source and third-party projects designed to help you tackle various aspects of your big data projects. But most are geared toward programmers, administrators, and technical professionals with specific skills.

What if you want to make big data accessible to business analysts, line-of-business leaders, and other personnel who aren't programmers? BigSheets is worth a look. It's a spreadsheet-style tool included with InfoSphere BigInsights that enables non-programmers to iteratively explore, manipulate, and visualize data stored in your distributed file system. Sample applications provided with BigInsights help you collect

and import data from various sources. In this article, we'll introduce you to BigSheets and two sample applications that complement it.

Background

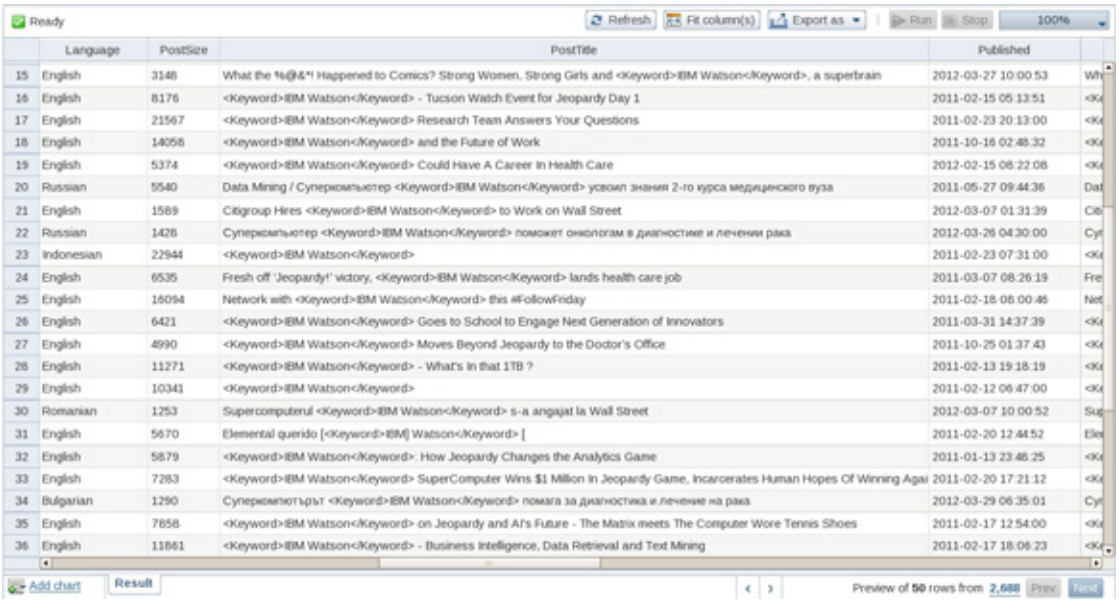
BigInsights is a software platform that can help companies discover and analyze business insights hidden in large volumes of a diverse range of data — data often ignored or discarded because it's too impractical or difficult to process using traditional means.

To help businesses efficiently derive value from such data, the Enterprise Edition of BigInsights includes several open source projects, including Apache Hadoop, and a number of IBM-developed technologies, including BigSheets. Hadoop and its related projects provide an effective software framework for data-intensive applications that exploit distributed computing environments to achieve high scalability.

IBM technologies enrich this open source framework with analytical software, enterprise software integration, platform extensions, and tools. For more information about BigInsights, see Resources. BigSheets is a browser-based analytic tool initially developed by IBM's Emerging Technologies group. Today, BigSheets is included with BigInsights to enable business users and non-programmers to explore and analyze data in distributed file systems. BigSheets presents a spreadsheet-like interface so users can model, filter, combine, explore, and chart data collected from various sources. The BigInsights web console includes a tab at top to access BigSheets. See Resources for further details on the web console.

Figure 1 depicts a sample data collection in BigSheets. While it looks like a typical spreadsheet, this collection contains data from blogs posted to public websites, and analysts can even click on links included in the collection to visit the site that published the source content.

Figure 1. Sample BigSheets collection based on social media data, with links to source content



	Language	PostSize	PostTitle	Published
15	English	3148	What the %@&*! Happened to Comics? Strong Women, Strong Girls and <Keyword>IBM Watson</Keyword>, a superbrain	2012-03-27 10:00:53
16	English	8176	<Keyword>IBM Watson</Keyword> - Tucson Watch Event for Jeopardy Day 1	2011-02-15 05:13:51
17	English	21567	<Keyword>IBM Watson</Keyword> Research Team Answers Your Questions	2011-02-23 20:13:00
18	English	14058	<Keyword>IBM Watson</Keyword> and the Future of Work	2011-10-16 02:48:32
19	English	5374	<Keyword>IBM Watson</Keyword> Could Have A Career In Health Care	2012-02-15 08:22:08
20	Russian	5540	Data Mining / Суперкомпьютер <Keyword>IBM Watson</Keyword> усвоил знания 2-го курса медицинского вуза	2011-05-27 09:44:36
21	English	1589	Citigroup Hires <Keyword>IBM Watson</Keyword> to Work on Wall Street	2012-03-07 01:31:39
22	Russian	1428	Суперкомпьютер <Keyword>IBM Watson</Keyword> поможет онкологам в диагностике и лечении рака	2012-03-26 04:30:00
23	Indonesian	22944	<Keyword>IBM Watson</Keyword>	2011-02-23 07:31:00
24	English	6535	Fresh off 'Jeopardy!' victory, <Keyword>IBM Watson</Keyword> lands health care job	2011-03-07 08:26:19
25	English	16094	Network with <Keyword>IBM Watson</Keyword> this #FollowFriday	2011-02-18 08:00:46
26	English	6421	<Keyword>IBM Watson</Keyword> Goes to School to Engage Next Generation of Innovators	2011-03-31 14:37:39
27	English	4990	<Keyword>IBM Watson</Keyword> Moves Beyond Jeopardy to the Doctor's Office	2011-10-25 01:37:43
28	English	11271	<Keyword>IBM Watson</Keyword> - What's in that 1TB ?	2011-02-13 19:18:19
29	English	10341	<Keyword>IBM Watson</Keyword>	2011-02-12 06:47:00
30	Romanian	1253	Supercomputerul <Keyword>IBM Watson</Keyword> s-a angajat la Wall Street	2012-03-07 10:00:52
31	English	5670	Elemental querido [<Keyword>IBM] Watson</Keyword> [	2011-02-20 12:44:52
32	English	5879	<Keyword>IBM Watson</Keyword>: How Jeopardy Changes the Analytics Game	2011-01-13 23:48:25
33	English	7283	<Keyword>IBM Watson</Keyword> SuperComputer Wins $1 Million In Jeopardy Game, Incarcerates Human Hopes Of Winning Again	2011-02-20 17:21:12
34	Bulgarian	1290	Суперкомпютърът <Keyword>IBM Watson</Keyword> помага за диагностика и лечение на рака	2012-03-29 06:35:01
35	English	7858	<Keyword>IBM Watson</Keyword> on Jeopardy and AI's Future - The Matrix meets The Computer Wore Tennis Shoes	2011-02-17 12:54:00
36	English	11861	<Keyword>IBM Watson</Keyword> - Business Intelligence, Data Retrieval and Text Mining	2011-02-17 18:06:23

After defining a BigSheets collection, an analyst can filter or transform its data as desired. Behind the scenes, BigSheets translates user commands, expressed through a graphical interface, into Pig scripts executed against a subset of the underlying data. In this manner, an analyst can iteratively explore various transformations efficiently. When satisfied, the user can save and run the collection, which causes BigSheets to initiate MapReduce jobs over the full set of data, write the results to the distributed file system, and display the contents of the new collection. Analysts can page through or manipulate the full set of data as desired.

Complementing BigSheets are a number of ready-made sample applications that business users can launch from the BigInsights web console to collect data from websites, relational database management systems (RDBMSes), remote file systems, and other sources. We'll rely on two such applications for the work described here. However, it's important to realize that programmers and administrators can use other BigInsights technologies to collect, process, and prepare data for subsequent analysis in BigSheets. Such technologies include Jaql, Flume, Pig, Hive, MapReduce applications, and others.

IBM Watson

IBM Watson is a research project that performs complex analytics to answer questions presented in a natural language. Watson's software consults data collected from various sources and uses Hadoop to efficiently process this data over a cluster of IBM Power 750 servers. IBM Watson placed first in a televised game show competition in 2011, beating two leading human contestants. See the Resources section for further details on IBM Watson and the Jeopardy! game show.

Before getting started, let's review the sample application scenario. It involves analyzing social media data about IBM Watson and, ultimately, joining this data with

simulated IBM internal data about media outreach efforts extracted from a relational DBMS. The idea is to explore the visibility, coverage, and "buzz" around a prominent brand, service, or project — a common requirement in many organizations. We won't cover the full range of analytic possibilities for such an application here, since our intent is simply to highlight how key aspects of BigSheets can help analysts get started quickly working with big data. But the work we'll explore will help you understand what's possible with little effort — and perhaps yield a surprise or two about IBM Watson's popularity.

Step 1: Gathering your data

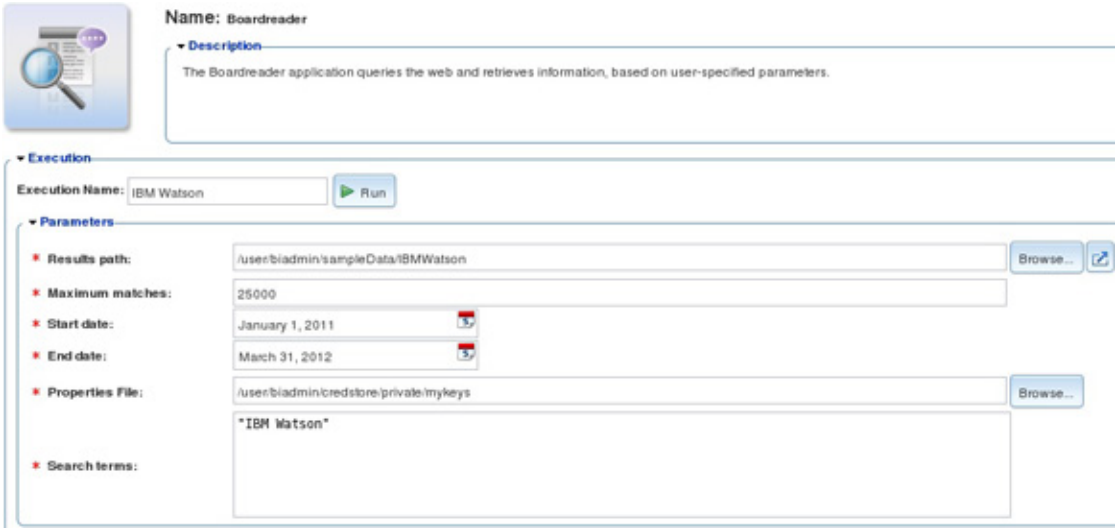
Before launching BigSheets, you need some data for your analysis. We'll focus first on collecting the social media data.

Collecting social media data

As you might expect, collecting and processing data extracted from various social media sites can be challenging, as different sites capture different information and employ different data structures. Furthermore, identifying and crawling a wide range of individual sites can be time-consuming.

Here, we used the BoardReader sample application provided with BigInsights to launch a search of blogs, news feeds, discussion boards, and video sites. Figure 2 illustrates the input parameters we provided to the BigInsights BoardReader application, which we launched from the Applications page of the BigInsights Web console. If you're not familiar with the web console and its catalog of sample applications, see the Resources section.

Figure 2. BoardReader application invocation from the BigInsights web console

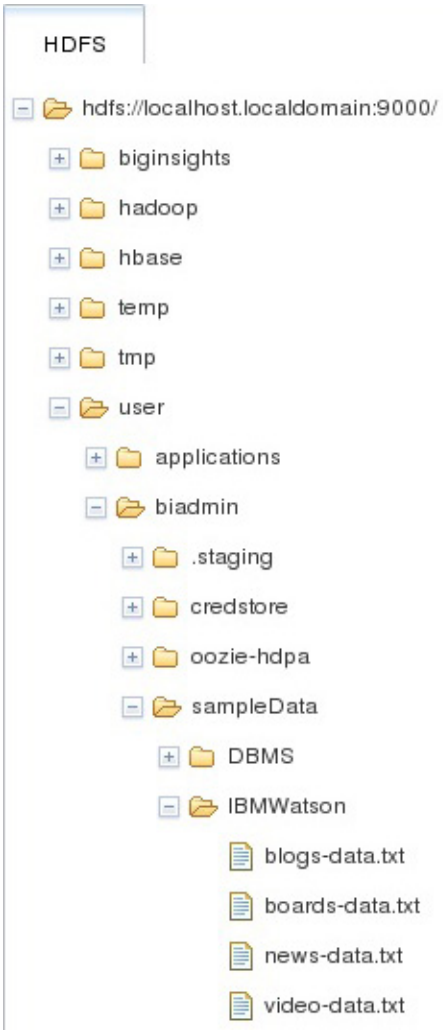


Let's briefly review the input parameters shown in Figure 2. The Results Path specifies the Hadoop distributed file system (HDFS) directory for the application's output. Subsequent parameters indicate that we restricted the returned results to

a maximum of 25,000 matches and the search period to 1 Jan 2011 through 31 Mar 2012. The Properties File references the BigInsights credentials store that we populated with our BoardReader license key. (Each customer must contact BoardReader to obtain a valid license key.) And "IBM Watson" is the subject of our search.

After running the application, the distributed file system contains four new files in the output directory, as shown at the bottom of Figure 3.

Figure 3. Application output stored in BigInsights



To keep things simple, we'll use news and blog data only in this article. If you want to follow along with our sample application scenario, execute the BoardReader application with the parameters we specified or download the sample data. Note that the download contains only a subset of the information BoardReader collects from blogs and news feeds. In particular, we removed the full-text/HTML content of posts and news items as well as certain metadata from the sample files. Such data isn't needed for the analytical tasks covered here, and we wanted to keep the size of each file manageable.

Each file returned by the BoardReader application is in JSON format. You can display a small portion of this data as text in the Files page of the BigInsights web console, but the results are difficult to read. In a moment, you'll see how to convert this data into "sheets" or BigSheets data collections, which are much easier to explore. But it's worth noting that each file contains a slightly different JSON structure — a situation to address when modeling a collection that unions the news and blog data sets. In big data projects, it's quite common to have to prepare or transform your data structures in some way to simplify subsequent analysis.

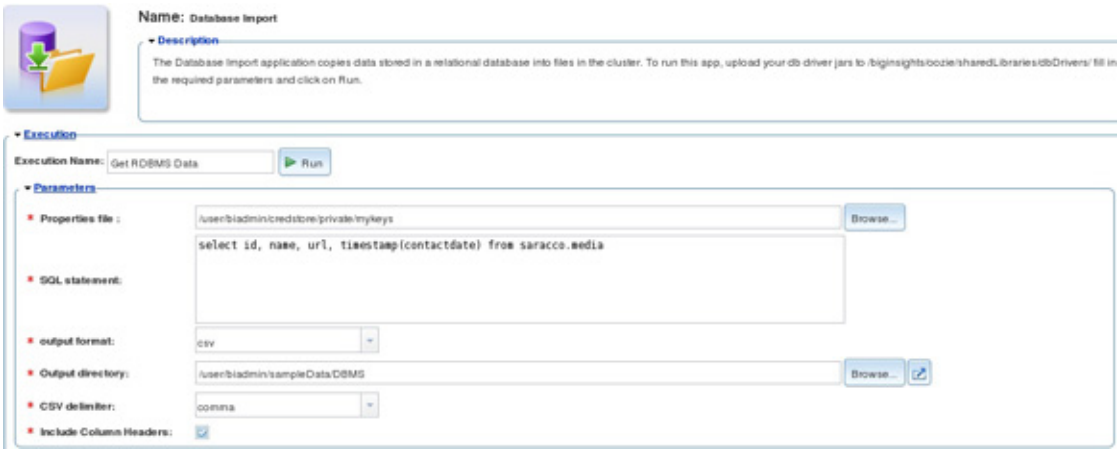
Collecting data from a relational DBMS

After exploring certain aspects of this social media data, we'll join it with data extracted from a relational DBMS. Many big data projects involve analyzing new information sources, such as social media data, in the context of existing enterprise information, including data stored in a relational DBMS. BigInsights provides connectivity to various relational DBMSes and data warehouses, including Netezza, DB2®, Informix®, Oracle, Teradata, and others.

For our sample scenario, we populated a DB2 table with simulated data about IBM media outreach efforts. Joining this relational data with information extracted from social media sites could give us some indication of the effectiveness and reach of various publicity efforts. While BigInsights provides dynamic relational DBMS query access through a command-line interface, we used the Data Import sample application of the BigInsights web console to extract the data of interest.

Figure 4 illustrates the input parameters we provided to this application. The mykeys properties file in the BigInsights credentials store contains the required JDBC input parameters for establishing a database connection, including the JDBC URL (for example, jdbc:db2://myserver.ibm.com:50000/sample), the JDBC driver class (for example, com.ibm.db2.jcc.DB2Driver), and the DBMS user ID and password. Other input parameters include a simple SQL SELECT statement to retrieve the data of interest from the target database, the output format (a comma-separated values file), and the BigInsights output directory for the results.

Figure 4. Data import application invocation from BigInsights web console



Note that prior to executing this application, we uploaded the appropriate DBMS driver files into the required BigInsights distributed file system directory (/biginsights/oozie/sharedLibraries/dbDrivers). Since DB2 Express-C was our source DBMS, we uploaded its db2jcc4.jar and db2jcc_license_cu.jar files.

To follow along with the DBMS-related work in our sample application scenario, obtain a free copy of DB2 Express-C (see Resources for a link), create and populate a sample table, and execute the BigInsights Data Import application as described. Alternatively, you can download the CSV file extracted from DB2 and upload it directly into BigInsights.

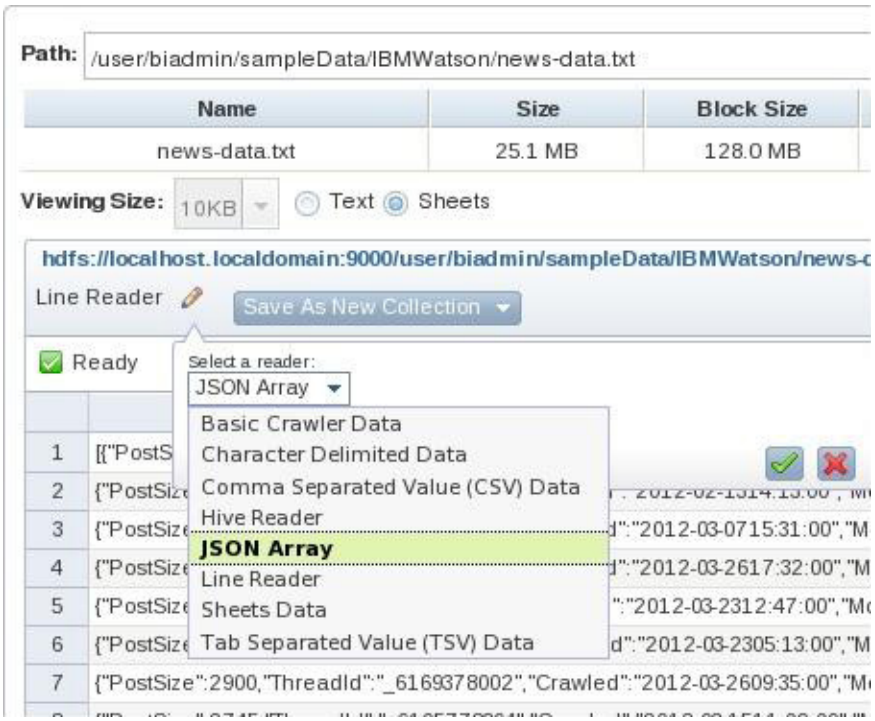
Step 2: Creating BigSheets collections

To begin analyzing your data with BigSheets, you need to create *collections* — spreadsheet-style structures — that model the files of interest in your distributed file system. For our scenario, these files include JSON-based blog data gathered by IBM's BoardReader application, JSON-based news data gathered by IBM's BoardReader application, and CSV-based data extracted from DB2 by IBM's Data Import application.

Let's step through the basics of creating one such collection:

1. From the Files page of the web console, use the file system navigator to select the news-data.txt file (see Figure 3).
2. In the right-side pane, select the Sheets button to change the display format from Text to Sheets. As shown in Figure 5, this button is located to the right of the Viewing Size specification.
3. Specify the appropriate "reader" or data format translator for your file. As Figure 5 also indicates, BigSheets provides various built-in readers for working with common data formats. For this sample file, the JSON Array reader is appropriate.
4. Save your new collection, naming it "Watson_news."

Figure 5. Creating a collection with an appropriate "reader"



Follow the same process to create a separate collection for the blogs-data.txt file, naming the collection "Watson_blogs." Finally, create a third collection for the CSV file with DBMS data, selecting the BigSheets Comma-Separated Values (CSV) Data as the reader for this file. Name this collection "Media_Contacts."

It's worth noting that you can create a collection based on the contents of a directory, rather than a single file. To do so, use the file system navigator to identify the target directory, click the **Sheets** button in the right-hand pane and specify the appropriate reader to be applied to all the files in the directory. However, the application scenario described in this article calls for three separate collections, as described earlier.

Step 3: Tailoring your collections

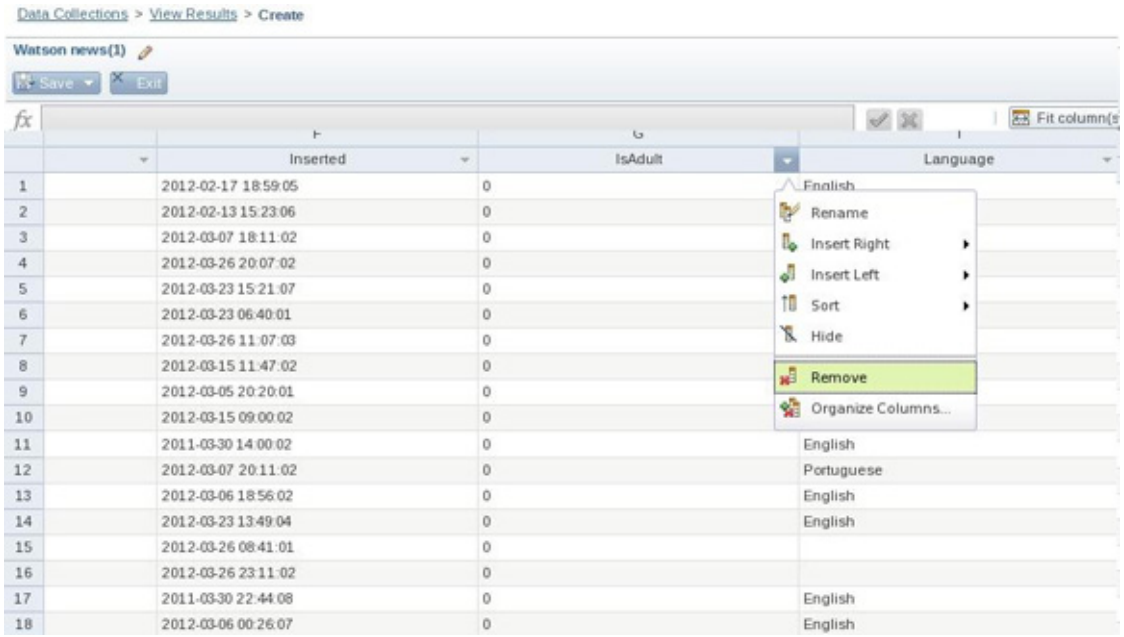
Quite often, analysts will want to tailor the format, content, and structure of their collections before investigating various aspects of the data itself. BigSheets provides a number of macros and functions to support such data preparation activities. In this section, we'll explore two such options: eliminating unnecessary data by deleting columns and consolidating data from two collections through a union operation.

Deleting columns

The BigInsights BoardReader application returns news and blog data that populate various columns in each BigSheets collection. We only need a subset of these columns for the analytical work we'll discuss in this article, so an important early step involves creating new collections that retain only the columns we want:

1. From the BigSheets main page, open the Watson_news collection you created from the news-data.txt file.
2. Click **Build New Collection**.
3. Navigate to the IsAdult column, as shown in Figure 6. Click the down arrow in the column heading and **Remove** the column. Do this for all columns in the collection except Country, FeedInfo, Language, Published, SubjectHtml, Tags, Type, and Url.
4. Save and exit, naming the new collection "Watson_news_revised." When prompted, run the collection. Note that a status bar to the right of the Run button enables you to monitor the progress of job. (Behind the scenes, BigSheets executes Pig scripts that initiate MapReduce jobs when you run a collection. As you might imagine, runtime performance depends on the volume of data associated with your collection and available system resources.)

Figure 6. Removing a column from a collection



Since we ultimately want to consolidate blog and news data into a single collection for further analysis, follow the same approach to create a new collection of blog data that contains only columns for Country, FeedInfo, Language, Published, SubjectHtml, Tags, Type, and Url. Name the new blog collection "Watson_blogs_revised."

Merging two collections into one through a union operation

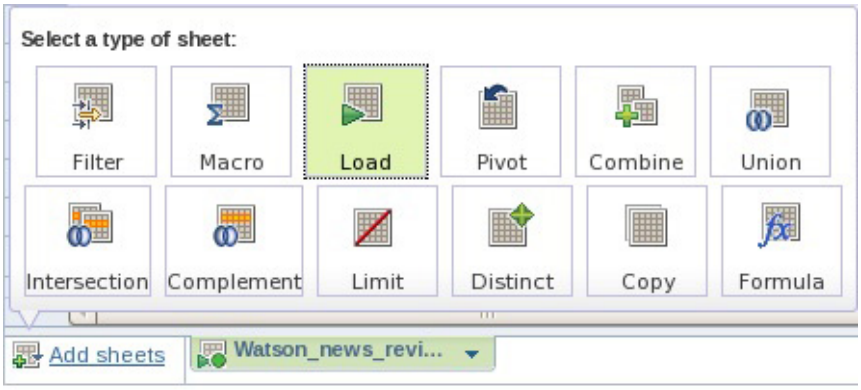
Next, merge the newly edited collections (Watson_news_revised and Watson_blogs_revised) into a single collection that will serve as the basis for exploring coverage of IBM Watson. To do so, use the BigSheets union operator. Note that it requires all sheets to have the same structure. If you followed the instructions in the prior section, you'll have two such collections to merge, each with Country,

FeedInfo, Language, Published, SubjectHtml, Tags, Type, and Url columns, in that order.

To merge the collections:

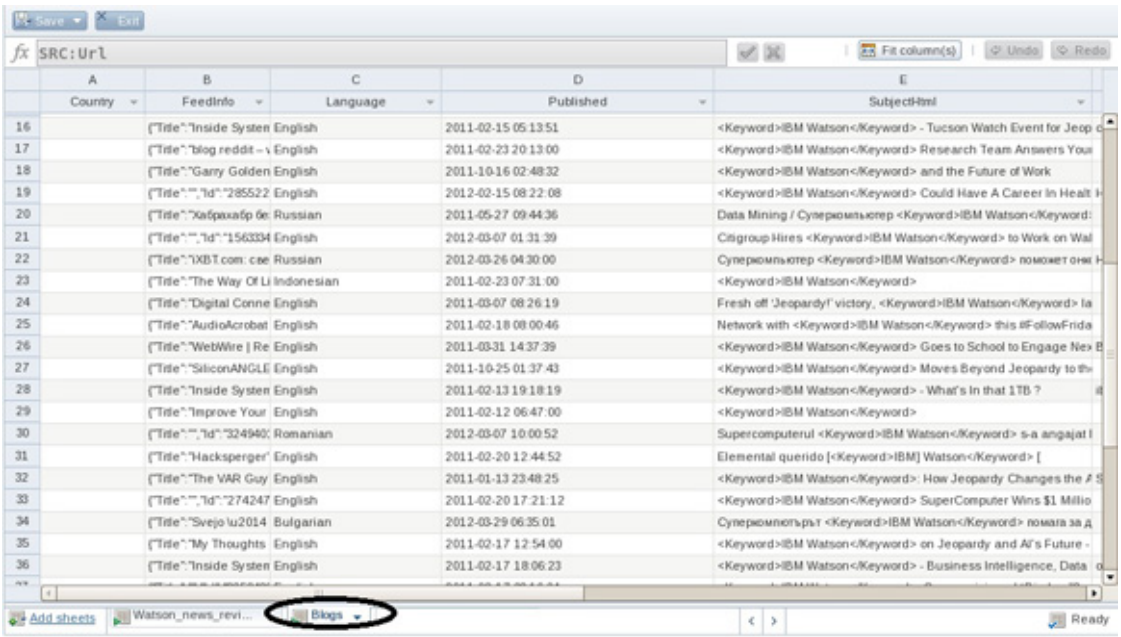
1. Open the Watson_news_revised collection and click **Build New Collection**.
2. Click **Add sheets > Load** to add the contents of another collection to your working model. (See Figure 7.) When prompted, select the **Watson_blogs_revised collection**, name your sheet "Blogs," and click the green check mark to apply the operation.

   Figure 7. Preparing to load a collection into a new sheet



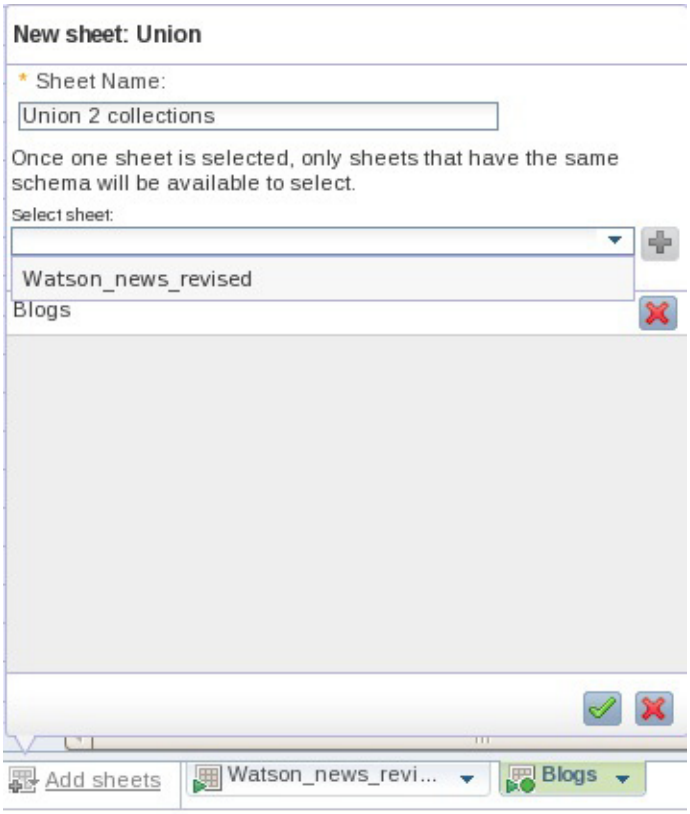
3. Inspect your display, which will contain the new sheet. Note that the lower left corner of your collection contains a new tab for it. (See Figure 8.)

   Figure 8. Reviewing a new sheet



4. Click **Add sheets > Union** to create another sheet to union the blog data with the news data. When prompted, click the drop-down menu and select **Watson_news_revised** as the sheet you will unite with the blog data you just loaded. (See Figure 9.) Click the plus sign (**+**) beside the box, then the green checkmark at bottom to initiate the union.

   Figure 9. Specifying sheets for union



5. Save and exit, naming it Watson_news_blogs. Run the collection.

Next, analyze the data in this new collection.

Step 4: Exploring a collection to investigate global coverage of IBM Watson

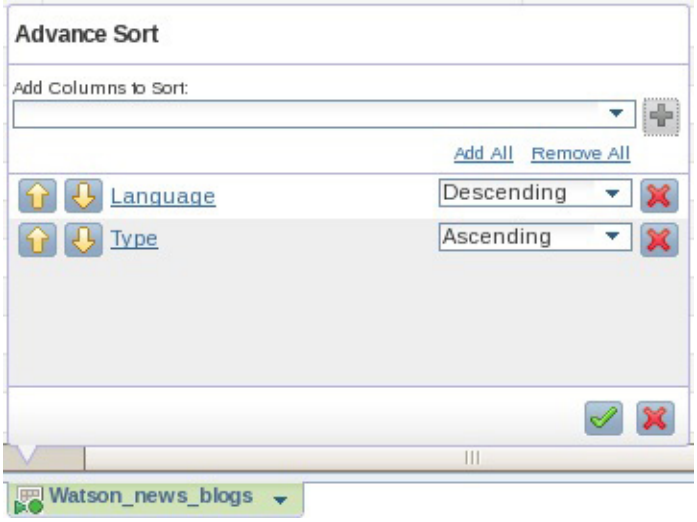
One area we'd like to explore involves global interest and coverage of IBM Watson. Initially, you might be tempted to sort the Watson_news_blogs collection on country column values. However, if you inspect the data, you'll note that many rows contain null values for this column. This is typical of data collected from social media sites and other sources. Often, desired data is missing, forcing analysts to consider other means to gain insight into areas of interest.

Sorting records

Most of our blog and news entries indicate the language of origin, so we'll sort our records by language and type to help us explore global coverage of IBM Watson in news and blog posts:

1. Open the Watson_news_blogs collection and click **Build New Collection**.
2. From the Language heading, expose the drop-down menu and click **Sort > Advanced**. When prompted, select the Language and Type columns from the Add Columns to Sort menu. Change Language's sort value to Descending and verify that Language is the primary sort column, as shown in Figure 10. Click the green arrow to apply the operation against a subset of your data.

   Figure 10. Preparing to sort a collection on two columns, with Language as the primary column



3. Inspect the sample 50 records displayed and note the various languages cited.
4. Save and exit your collection, naming it Watson_sorted. Then run the collection against the full data set. When you inspect the returned results, you'll see more records for specific languages, such as Vietnamese, than you did in the previous step.

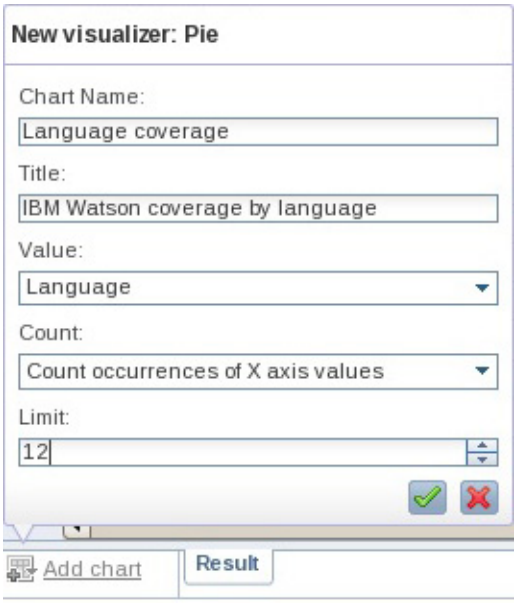
Charting results

Although you can page through your collection to explore coverage of IBM Watson in various languages, the easiest way to visualize worldwide "buzz" may be to chart the results. Doing so provides a broad view, which can serve as inspiration for further exploratory and analytical efforts. BigSheets supports a variety of chart types, including bar charts, pie charts, tag clouds, and others. We'll use a simple pie chart here:

1. With the Watson_sorted collection open, click **Add chart > Chart > Pie**. (The Add chart tab is in the lower-left corner of the collection beside the Result tab.)

2. When prompted, supply values of your choice for the chart name and title. Select the Language column as the value you want to chart, leaving the Count field set to its default value. Reset the Limit value to 12, so the pie chart will reflect data about the 12 most frequently occurring languages in this collection. See Figure 11.

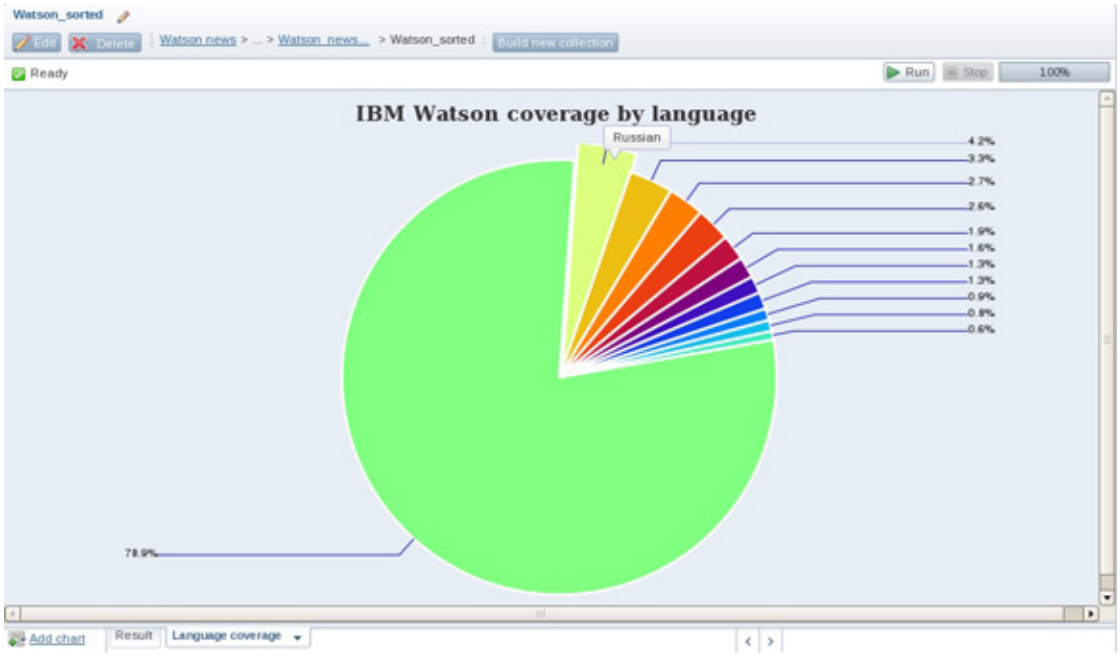
   Figure 11. Input parameters for creating a pie chart



3. Click the green checkmark and run the chart when prompted.

As you might expect, the resulting pie chart indicates that nearly 79 percent of the news and blog data we collected was published in English. But can you guess the next most popular language for IBM Watson? The pie chart illustrated in Figure 12 indicates that it's Russian. By hovering over any slice of a pie chart displayed in BigSheets, you can determine its underlying value (in this case, the Language column value).

Figure 12. Measuring global interest in IBM Watson by language, based on available news and blog data



Cleansing data values

If you hover over the fifth and sixth largest slices of the pie chart shown in Fig. 12 (with percentages 2.6 and 1.9), you'll find that they're both variations of Chinese. This illustrates another common situation involving data collected from various data sources, such as different social media sites — data values you might want to treat as identical are often represented somewhat differently.

Let's explore how to use BigSheets to alter these values so variations of Chinese are replaced by a single value of "Chinese":

1. If needed, open the Watson_sorted collection and click the **Edit** button (beneath the collection's name in the upper-left corner).
2. Navigate to the Language column and click the down arrow in the column heading to expose the drop-down menu. Select **Insert Right > New Column** to create a new column to hold the cleansed data. When prompted, name the new column LanguageRevised and click the green checkmark to complete the operation.
3. With your cursor positioned on the LanguageRevised column, enter the following formula in the fx (formula specification) box at the top of the sheet: `IF(SEARCH('Chin*', #Language) > 0, 'Chinese', #Language)`. See Figure 13.

Figure 13. Specifying a formula to derive a column's value

This formula causes BigSheets to search for values beginning with "Chin" in the Language column of the sheet. When it finds such values, it writes "Chinese" in the LanguageRevised column; otherwise, it copies the value found in the Language column into the LanguageRevised column. The BigInsights InfoCenter (included in the Resources section) contains details on defining BigSheets formulas. Click the green checkmark to apply the formula.

4. Save and exit your work. When a warning appears about the data being out of sync, run the revised definition of this collection.
5. Create a new 12-slice pie chart based on the values in the LanguageRevised column and compare the results to the pie chart you created earlier (based on "raw" data in the Language column). Note that your new pie chart shows that "Chinese" is the second most common language, followed by Russian, Spanish, and German.

Digging deeper: Filtering results and extracting URL data

The data you just inspected can provoke a range of questions that require further investigation. This is quite typical of big data analysis, which is often iterative and exploratory by nature. Let's dig a bit deeper into the coverage of IBM Watson in English-based news and blog posts to try to pinpoint coverage in the United Kingdom.

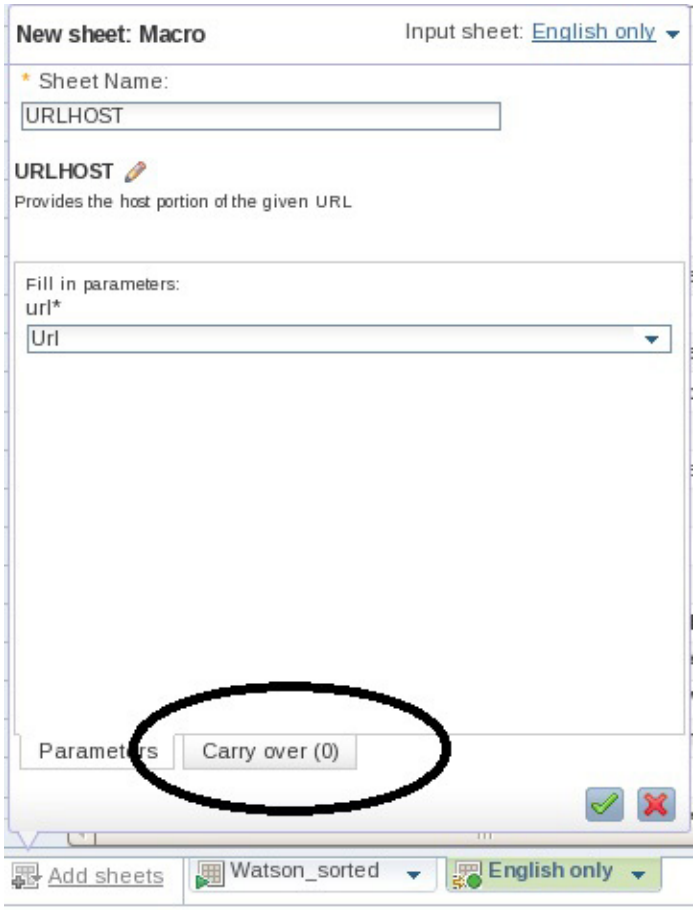
In keeping with the introductory nature of this article, we'll take a simple approach to investigating this topic. Specifically, we'll derive a new collection from the Watson_sorted collection that retains English-language records with URL domain names ending in ".uk" or a Country value of "GB" (for Great Britain). To achieve this, we'll need to use the BigSheets Filter operator as well as a macro for extracting URL host data from a full URL string:

1. Open the Watson_sorted collection and build a new collection.
2. Add a sheet that employs the Filter operation.
3. When prompted, select **Match all** and specify Language is English in the three drop-down menu boxes, as shown in Figure 14. Then click the green checkmark to apply the operation to a subset of the collection's data.

Figure 14. Filtering based on a column value

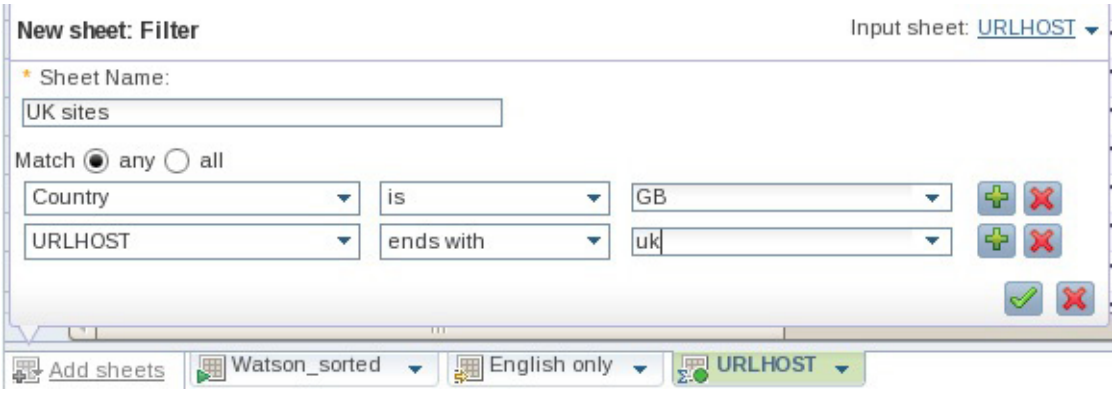
4. Save your work (name the sheet Watson_sorted_English_UK), but don't exit, as you'll continue to refine this collection.
5. Add another sheet that invokes a Macro. When prompted, click **Categories > url > URLHOST**. Select the URL column of your collection as the target column containing URL values. (The macro will read the values in this column and extract the URL host information from the larger string. For example, given a URL value of "http://www.georgeemsden.co.uk/2011/09/how-long-before-your-laptop-finds-a-cure-for-cancer/," the macro will return "www.georgeemsden.co.uk" as the URL host name.)
6. Click the **Carry Over** tab at the bottom of the pane, as shown in Figure 15. This is important because it enables you to specify which columns of the existing collection you want to retain (or "carry over").

Figure 15. Working with the URLHOST macro



7. Click **Add all** to retain all the existing columns and apply the operation. Save your work, but don't exit.
8. Add another sheet to Filter the data further. When prompted, match any of the following two criteria: "URLHOST ends with uk" and "Country is GB," as shown in Figure 16. (Given the sparse nature of the data in this collection, we need to match either of these conditions to detect U.K.-based URL host sites.) Apply the operation.

Figure 16. Filtering data based on two column values



9. Save and exit the collection, then run it.

Sorting the results on URLHOST column or plotting a chart will enable you to quickly identify which U.K. sites in the resulting collection most frequently covered IBM Watson. For example, Figure 17 depicts a tag cloud chart we generated for the top 10 such sites. As with any BigSheets tag cloud, larger font indicates more occurrences of the data value and scrolling over a data value reveals the number of times it occurred in the collection.

Figure 17. Top 10 U.K. sites with coverage of IBM Watson

Step 5: Investigating further, combining social media and structured data

Before concluding this introduction to BigSheets, let's explore a few other areas of interest involving our sample data set:

- The number of distinct sites covering IBM Watson and the top 12 worldwide sites with coverage of IBM Watson. To accomplish this, we'll introduce additional macros and another chart type.
- Coverage by sites that were the subject of IBM media outreach efforts. To accomplish this, we'll join data extracted from a relational database with social

media data in BigInsights. (For this article, we created fictitious data about IBM public relations work.)

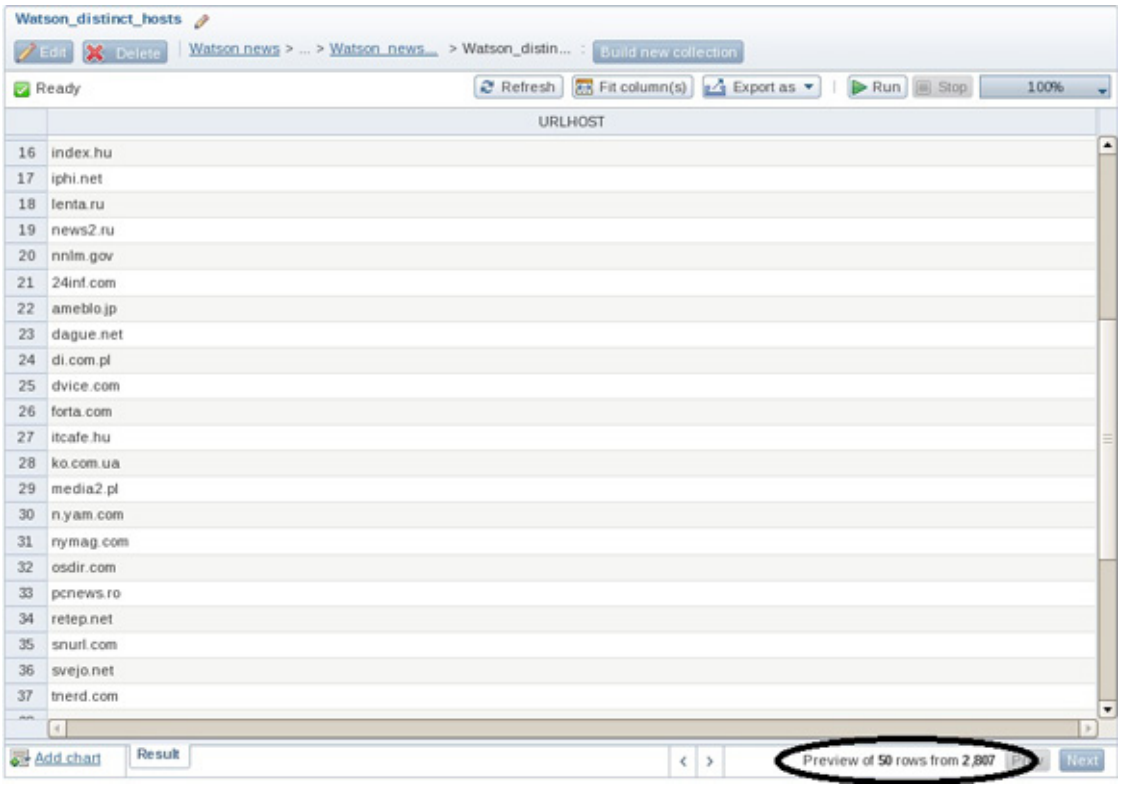
Finally, we'll discuss how to export the contents of a collection into a popular data format that can be easily used by third-party applications.

Determining the breadth of coverage and top 12 sites

One aspect of evaluating the effectiveness of a media outreach campaign involves assessing the breadth of coverage. In this example, you'll use BigSheets to determine the number of distinct news and blog sites with coverage of IBM Watson.

1. Open the Watson_news_blogs collection and build a new collection.
2. Add a sheet named "Url Hosts" that uses the URLHOST macro to extract the URL host name from the full string provided in the URL column. Carry over the URL column only. (If needed, refer to the instructions in Step 4 for details on the URLHOST macro.)
3. Add another sheet, applying the Distinct operator to the sheet you just created.
4. Save and exit this collection, running it when prompted. Observe that there are slightly more than 2,800 distinct sites, as shown in the lower-right corner of Figure 18. If you open the Watson_news_blogs collection, you'll see there are more than 7,200 total records.

Figure 18. Determining the number of distinct host sites



Now that you know that a number of sites contain multiple posts, you might want to identify the top 12 sites containing the most posts about IBM Watson and visualize the results in a bar chart. That's easy to do, and the results might even surprise you:

1. If necessary, open the collection you just created.
2. Click **Add chart > Chart > Column**. Provide values of your choice for chart's name and title. Retain the default values for the X and Y axes. Set the Limit to 12. Apply these settings and run the chart. Figure 19 illustrates the results. If you expected IBM or an IBM-sponsored site to be among the top three, that's not the case.

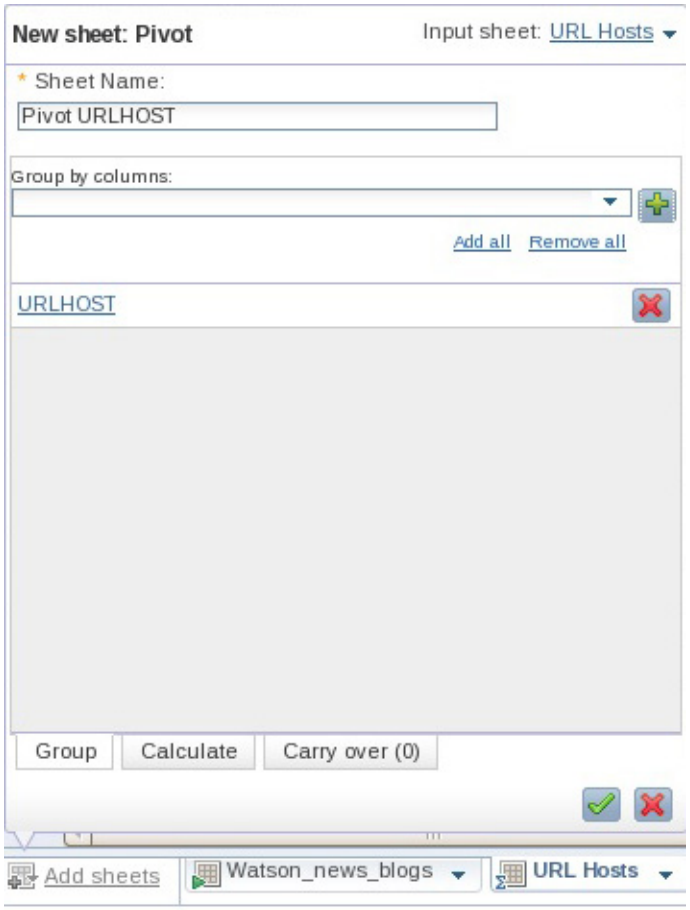
   Figure 19. Charting the top 12 sites covering IBM Watson based on number of posts

If you inspect the URLs for the top two sites, you'll see that they're variants of bizjournals.com, indicating that you may want to return to the collection and transform or cleanse this data. As mentioned, big data analytics often requires iterative exploration, processing, and refinement of data.

Finally, identifying the Top 12 sites might make you curious about the number of posts for each URL host site. Let's wrap this example by implementing an easy approach to obtaining that information:

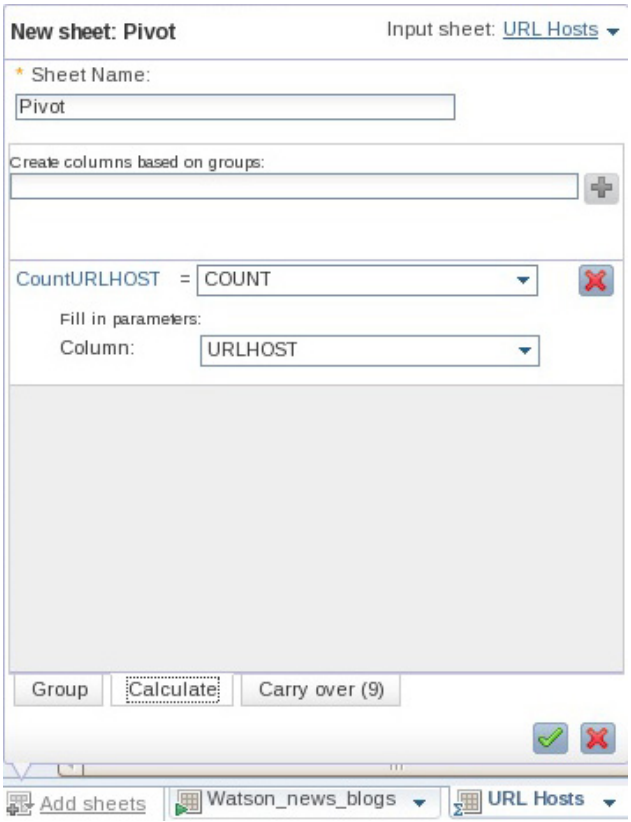
1. If needed, open the collection and edit it.
2. Click **Add Sheet > Pivot**. Name the sheet "Pivot," identify the URL hosts sheet as the input sheet, and select **URLHOST** as the pivot column. See Figure 20.

Figure 20. Creating a Pivot sheet to hold aggregated data



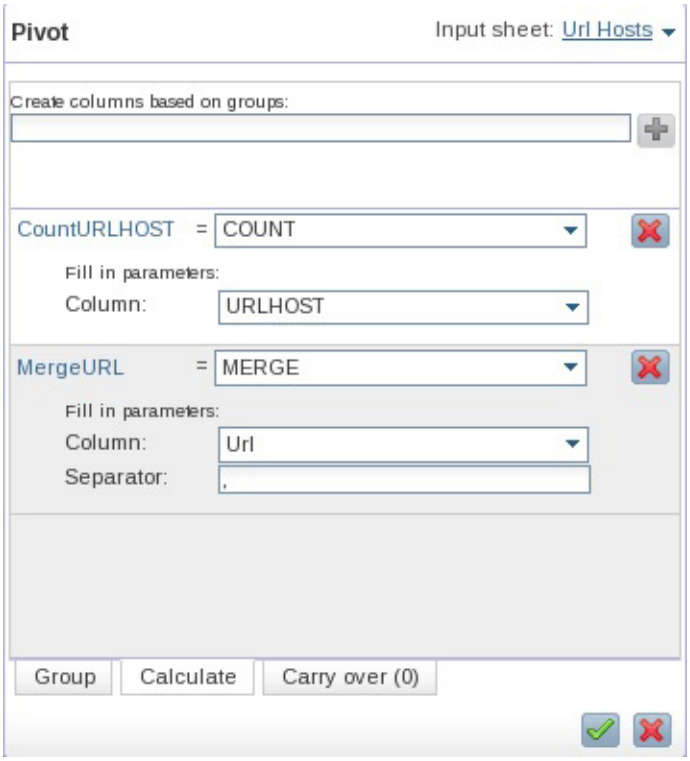
3. Click the **Calculate** tab at the bottom of the menu. Specify the name of a new column to contain the aggregated data (CountURLHOST, for example) and click the plus sign (+). For the new column's value, select **COUNT** and identify URLHOST as the target column for the count operation. (See Figure 21.)

Figure 21. Specifying initial calculation parameters for your new Pivot sheet



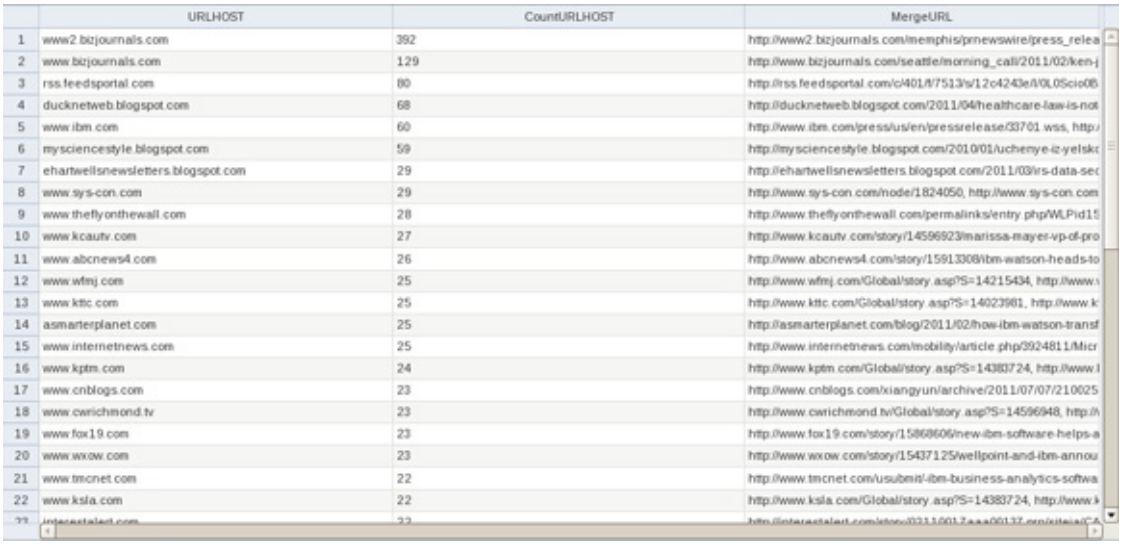
4. While still on the Calculate tab, create another column named MergeURL to hold the merged list of full URLs associated with the URLHOST values in the first column of your collection. Such a list may come in handy later. To generate this list and include it as a new column in the resulting collection, click the plus sign, select MERGE for the new column's value, Url as the target column, and a comma (,) as the field separator. Verify that your calculation specification appears like Figure 22 and apply the operation.

Figure 22. Adding a second calculation to your Pivot sheet



5. If desired, sort the values in the aggregated column (CountURLHOST) in descending order.
6. Save and exit the collection, then run it. Browse through the results, a subset of which are shown in Figure 23.

Figure 23. Inspecting aggregated data contained in a Pivot sheet



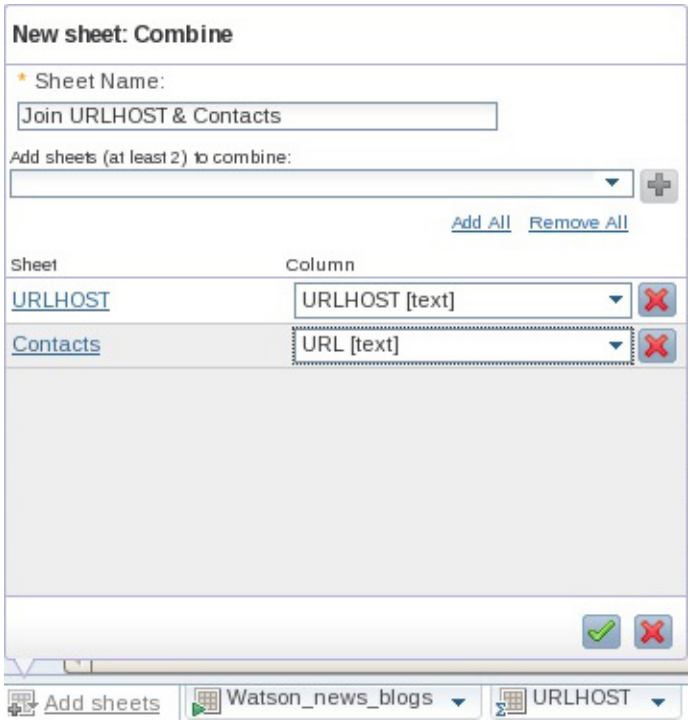
	URLHOST	CountURLHOST	MergeURL
1	www2.bizjournals.com	392	http://www2.bizjournals.com/memphis/prnewswire/press_relea
2	www.bizjournals.com	129	http://www.bizjournals.com/seattle/morning_call/2011/02/ken-j
3	rss.feedsportal.com	80	http://rss.feedsportal.com/c/401/f/7513/s/12c4243e/l/0L0Scio0B
4	ducknetweb.blogspot.com	68	http://ducknetweb.blogspot.com/2011/04/healthcare-law-is-not
5	www.ibm.com	60	http://www.ibm.com/press/us/en/pressrelease/33701.wss, http:/
6	mysciencestyle.blogspot.com	59	http://mysciencestyle.blogspot.com/2010/01/uchenye-iz-yelsk
7	ehartwellsnewsletters.blogspot.com	29	http://ehartwellsnewsletters.blogspot.com/2011/03/irs-data-sec
8	www.sys-con.com	29	http://www.sys-con.com/node/1824050, http://www.sys-con.com
9	www.theflyonthewall.com	28	http://www.theflyonthewall.com/permalinks/entry.php/WLPid15
10	www.kcautv.com	27	http://www.kcautv.com/story/14596923/marissa-mayer-vp-of-pro
11	www.abcnews4.com	26	http://www.abcnews4.com/story/15913308/ibm-watson-heads-to
12	www.wfmj.com	25	http://www.wfmj.com/Global/story.asp?S=14215434, http://www.w
13	www.kttc.com	25	http://www.kttc.com/Global/story.asp?S=14023981, http://www.k
14	asmarterplanet.com	25	http://asmarterplanet.com/blog/2011/02/how-ibm-watson-transf
15	www.internetnews.com	25	http://www.internetnews.com/mobility/article.php/3924811/Micr
16	www.kptm.com	24	http://www.kptm.com/Global/story.asp?S=14383724, http://www.k
17	www.cnblogs.com	23	http://www.cnblogs.com/xiangyun/archive/2011/07/07/210025
18	www.cwrichmond.tv	23	http://www.cwrichmond.tv/Global/story.asp?S=14596948, http://w
19	www.fox19.com	23	http://www.fox19.com/story/15868606/new-ibm-software-helps-a
20	www.wxow.com	23	http://www.wxow.com/story/15437125/wellpoint-and-ibm-annou
21	www.tmcnet.com	22	http://www.tmcnet.com/usubmit/-ibm-business-analytics-softwa
22	www.ksla.com	22	http://www.ksla.com/Global/story.asp?S=14383724, http://www.k
23	interestalert.com	22	http://interestalert.com/story/0311/0017aaa00137.rm/sites/CA

Correlating internal media outreach efforts with external coverage

Until now, our BigSheets work has involved only data collected from external sites. However, many big data projects call for combining external data with internal corporate data, such as data in a relational DBMS. In this section, you'll use BigSheets to join two collections: one modeling social media data and one modeling relational data. By joining these two collections, you'll be able to explore how corporate media outreach efforts correlate to coverage by third-party websites. Note that the sample relational data we provide as a CSV file with this article contains simulated information about IBM media contacts. Here's how to join the collections and visualize the results:

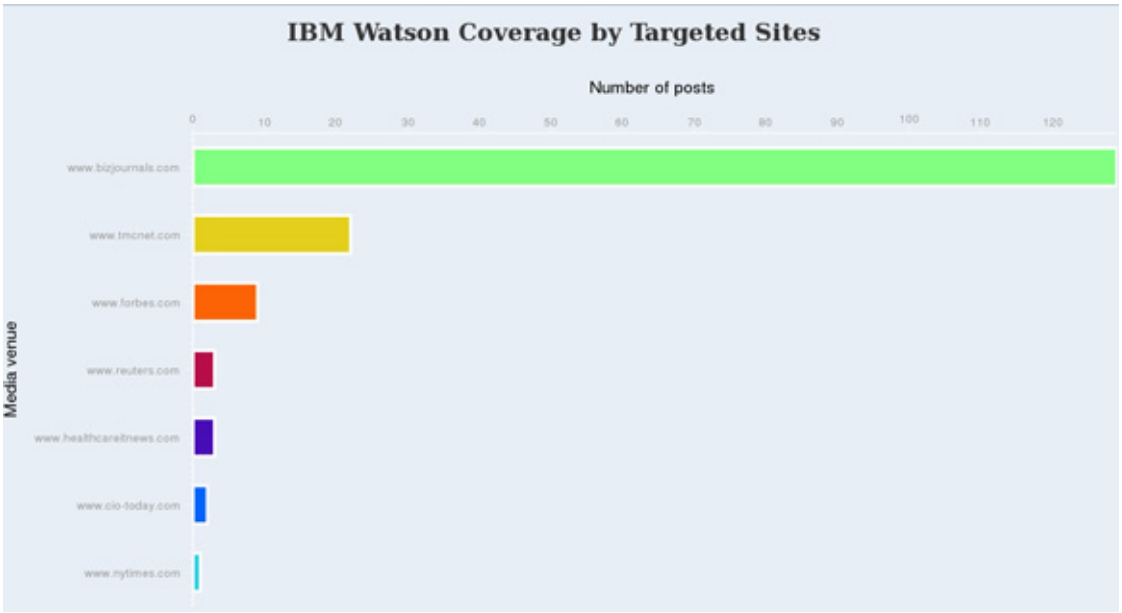
1. Open the Watson_news_blogs collection and build a new collection.
2. Add a sheet using the URLHOST macro to extract host name information. Carry over all existing columns and name the sheet URLHOST.
3. Add another sheet that loads the Media_Contacts collection you built earlier based on the imported RDBMS data. (You created this collection in Step 2.) Name this new sheet Contacts.
4. Rename the final column of the Contacts sheet to LastContact. (This column was generated by invoking the `SQL TIMESTAMP()` function against the original RDBMS data. Its values indicate when the target media provider was last contacted.)
5. Add another sheet that combines the URLHOST and Contacts sheets based on the values of the URLHOST and URL columns, respectively (see Figure 24). Name this new sheet Combine.

   Figure 24. Combining (joining) data from two sheets



6. To make it easier to inspect the results, delete the ID and URL columns that originated from the Media_Contacts sheet. Reorganize the remaining columns so they appear in a more intuitive order, such as URLHOST, NAME, Published, LastContact, FeedInfo, Country, Language, SubjectHtml, Tags, Type, Url.
7. Save the collection and run it. Skim through the results, or chart them (if desired) to assess the volume of posts for each targeted media site. (Figure 25 depicts a horizontal bar chart summarizing this data.)

Figure 25. Assessing number of posts about IBM Watson at various sites



Exporting your collection

In some cases, the results of your BigSheets analysis may be useful to downstream applications or colleagues who aren't authorized to work directly with BigInsights. Fortunately, it's easy to export one or more of your collections into popular data formats. Simply open the target collection and use the **Export As** function (to the left of the **Run** button) and select JSON, CSV, ATOM, RSS, or HTML as the target format. The results will be displayed in your browser, and you can save the output to your local file system.

A peek beyond the basics

By now, you have some idea of what BigSheets can do. Hopefully, you've seen how built-in macros, functions, and operators enable you to explore, transform, and analyze various forms of big data without writing code in Java™ or scripting language.

While we kept our scenario simple to get you up to speed quickly on the basics of BigSheets, there's more to this technology — and complementary BigInsights technologies — than we can cover in an introductory article. For example, many

social media analytics projects require digging into the content of posts to assess sentiment, categorize content, eliminate false positives, etc. Such efforts require extracting context from textual data, a capability offered through another component of BigInsights that will be the subject of a future article. Fortunately, such text analytical capabilities can be combined with BigSheets through custom plug-ins.

In addition, certain analytical tasks may require a query language to easily express various conditions, process and transform nested data structures, apply complex conditional logic constructs, etc. Indeed, BigInsights includes Jaql, a JSON-based query language, that programmers often use to read and prepare data for subsequent analysis in BigSheets. A future article will explore Jaql.

Summary

This article explored how BigInsights enables business analysts to work with big data without writing code or scripts. In particular, it introduced two sample applications for gathering social media and RDBMS data and explained how analysts can model, manipulate, analyze, combine, and visualize this data using BigSheets, a spreadsheet-style tool designed for business analysts. To keep things simple, this article explored a subset of BigSheets operators and functions, concentrating on those most relevant to our sample application scenario involving coverage of IBM Watson, a research project that uses Apache Hadoop to perform complex analytics to answer questions presented in a natural language.

If you're ready to get started with a big data project, see the Resources section for links to software downloads, online education, and other materials related to BigInsights.

Acknowledgements

Special thanks to Stephen Dodd, vice president at Effyis Inc., for authorizing us to make sample BoardReader output data available for download with this article. Thanks also to IBM's Diana Pupons-Wickham and Gary Robinson for reviewing this article.

Downloads

Description	Name	Size	Download method
Sample social media data and relational data	sampleData.zip	1030KB	HTTP

Information about download methods

Resources

Learn

- Read "Understanding InfoSphere BigInsights" to learn more about the product's architecture and underlying technologies.
- Watch the Big Data: Frequently Asked Questions for IBM InfoSphere BigInsights video to listen to Cindy Saracco discuss some of the frequently asked questions about IBM's Big Data platform and InfoSphere BigInsights.
- Watch Cindy Saracco demonstrate portions of the scenario described in this article in Big Data -- Analyzing Social Media for Watson.
- Read "Exploring your InfoSphere BigInsights cluster and sample applications" to learn more about the product's web console.
- Watch Anshul Dawra or Cindy Saracco explain BigSheets in one or more of these videos: Big Data Patent Data Analysis with BigSheets, Big Data for Business Users — an introduction to BigSheets for InfoSphere BigInsights, and Big Data — BigSheets in Action.
- Visit the BigInsights Technical Enablement wiki for links to technical materials, demos, training courses, news items, and more.
- Learn about the IBM Watson research project and its post-Jeopardy! activities.
- Check out Big Data University for free courses on Hadoop and big data.
- Refer to the IBM InfoSphere BigInsights Information Center for product documentation.
- Order a copy of *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data* for details on two of IBM's key big data technologies.
- Learn more about Information Management at the developerWorks Information Management zone. Find technical documentation, how-to articles, education, downloads, product information, and more.
- Stay current with developerWorks technical events and webcasts.
- Follow developerWorks on Twitter.

Get products and technologies

- Try an evaluation copy of IBM InfoSphere BigInsights Basic Edition.
- Build your next development project with IBM trial software, available for download directly from developerWorks.
- Now you can use DB2 for free. Download DB2 Express-C, a no-charge version of DB2 Express Edition for the community that offers the same core data features as DB2 Express Edition and provides a solid base to build and deploy applications.

Discuss

- Check out the developerWorks blogs and get involved in the developerWorks community.

About the authors

Cynthia M. Saracco

Cynthia M. Saracco works on database management and XML technologies at IBM's Silicon Valley Lab. She has co-authored three books and taught university-level courses on various software technologies.

Anshul Dawra

Anshul Dawra is a Senior Software Engineer in the IBM Information Management group at Silicon Valley Labs in San Jose, CA. He is an architect in the pureQuery and Extended Insight team. Before joining the pureQuery team, he worked on the design and development of IBM Data Server Driver for JDBC and SQLJ.