Himanshu Choudhary
Edward He
Cong Kim
Matt Miller
Professor Roger Chiang
IS4030 Business Intelligence

MongoDB and Tableau

Project Scopes

For this project, our goal was to fetch data from a web based API, store the unstructured data in MongoDB, and connect that data to Tableau for visualization and analysis. The challenge in this is that historically, Tableau does not work well with unstructured data. In the past, analyst and programmers would use ETL tools to create a traditional relational database from their unstructured data. Next, they would connect Tableau to that intermediate database. Finally, they would be able to create visualizations and analysis with that data. The result is that they are often not working with the most current data that they have available. They would not have real-time connectivity with their data. MongoDB and Tableau have recently been working together to make the integration of their platforms easier.

In this report, we will discuss the steps that we took to import data into MongoDB, connect MongoDB with Tableau, and create visualizations and analysis of our data in Tableau. Within the report, the terms JavaScript Object Notation (JSON) and Binary JSON (BSON) are used interchangeably. BSON is a superset of JSON and is the document type used in MongoDB.

Resources

MongoDB and Tableau have great knowledge bases and community forums. We were able to find the answers to questions that we had about these platforms using those resources. The direct integration of MongoDB with Tableau however, has only recently been possible. While there is an abundance of information available about these products, instructions and advice on their integration is scarce.

Dataset

To begin, we needed data. We considered using a sample dataset like Microsoft's Northwind or AdventureWorks, however we decided to challenge ourselves with a more complex project. Simulating a scenario similar to what we anticipate encountering in industry, and inspired by Sam McFarland's demonstration in class, we harvested data from a live data stream. We chose Weather Underground's (www.wunderground.com) as our data source.

Using Weather Underground's application program interface (API), we collected data from one weather station in San Francisco. The API allows for up to 500 calls per day and 10 calls per minute, at no charge. For a monthly fee, higher volumes of calls can be purchased. For our project, we chose to stay within the limits of the free quota. We retrieved data every hour, for 11 days.

API requests are made using a url request to the server, which can be tested at Weather Underground. The key to access has to be applied for. Requests are made over HTTP and return JSON data. Below is an example of a web request:

http://api.wunderground.com/api/88f2e9a1a85ff5d4/conditions/q/CA/San_Francisco.json

Below is an example of a JSON document that was returned from a request.

```
/* 1 */
{
    "_id" : ObjectId("56f8816ccd574b2a74c059b8"),
    "response" : {
        "version" : "0.1",
        "termsofService" : "http://www.wunderground.com/weather/api/d/terms.html",
        "features" : {
            "conditions" : 1
        }
    },
    "current_observation" : {
        "image" : {
            "url" : "http://icons.wxug.com/graphics/wu2/logo_130x80.png",
            "title" : "Weather Underground",
            "link" : "http://www.wunderground.com"
        },
        "display_location" : {
            "full" : "San Francisco, CA",
    "city" : "San Francisco",
    "state" : "CA",
    "state_name" : "California",
    "country" : "US",
    "country_iso3166" : "US",
    "zip" : "94101",
    "magic" : "1",
    "wmo" : "99999",
    "latitude" : "37.77500916",
    "longitude" : "-122.41825867",
    "elevation" : "47.00000000"
},
"observation_location" : {
    "full" : "SOMA - Near Van Ness, San Francisco, California",
    "city" : "SOMA - Near Van Ness, San Francisco",
    "state" : "California",
    "country" : "US",
    "country_iso3166" : "US",
    "latitude" : "37.773285",
    "longitude" : "-122.417725",
    "elevation" : "49 ft"
},
"estimated" : {},
"station_id" : "KCASANFR58",
"observation_time" : "Last Updated on March 27, 5:56 PM PDT",
"observation_time_rfc822" : "Sun, 27 Mar 2016 17:56:32 -0700",
"observation_epoch" : "1459126592",
"local_time_rfc822" : "Sun, 27 Mar 2016 17:57:15 -0700",
"local_epoch" : "1459126635",
"local_tz_short" : "PDT",
"local_tz_long" : "America/Los_Angeles",
"local_tz_offset" : "-0700",
"weather" : "Partly Cloudy",
"temperature_string" : "58.3 F (14.6 C)",
"temp_f" : 58.3,
"temp_c" : 14.6,
"relative_humidity" : "69%",
"wind_string" : "From the WSW at 4.0 MPH Gusting to 13.0 MPH",
"wind_dir" : "WSW",
"wind_degrees" : 241,
"wind_mph" : 4,
"wind_gust_mph" : "13.0",
"wind_kph" : 6.4,
"wind_gust_kph" : "20.9",
        "pressure_mb" : "1018",
        "pressure_in" : "30.07",
        "pressure_trend" : "0",
        "dewpoint_string" : "48 F (9 C)",
        "dewpoint_f" : 48,
        "dewpoint_c" : 9,
        "heat_index_string" : "NA",
        "heat_index_f" : "NA",
        "heat_index_c" : "NA",
        "windchill_string" : "NA",
        "windchill_f" : "NA",
        "windchill_c" : "NA",
        "feelslike_string" : "58.3 F (14.6 C)",
        "feelslike_f" : "58.3",
        "feelslike_c" : "14.6",
        "visibility_mi" : "10.0",
        "visibility_km" : "16.1",
        "solarradiation" : "--",
        "UV" : "1",
        "precip_1hr_string" : "0.00 in ( 0 mm)",
        "precip_1hr_in" : "0.00",
        "precip_1hr_metric" : " 0",
        "precip_today_string" : "0.00 in (0 mm)",
        "precip_today_in" : "0.00",
        "precip_today_metric" : "0",
        "icon" : "partlycloudy",
        "icon_url" : "http://icons.wxug.com/i/c/k/partlycloudy.gif",
        "forecast_url" : "http://www.wunderground.com/US/CA/San_Francisco.html",
        "history_url" :
"http://www.wunderground.com/weatherstation/WXDailyHistory.asp?ID=KCASANFR5
8",
        "ob_url" : "http://www.wunderground.com/cgi-
bin/findweather/getForecast?query=37.773285,-122.417725",
        "nowcast" : ""
    }
}
```

Microsoft Visual Studio and C#

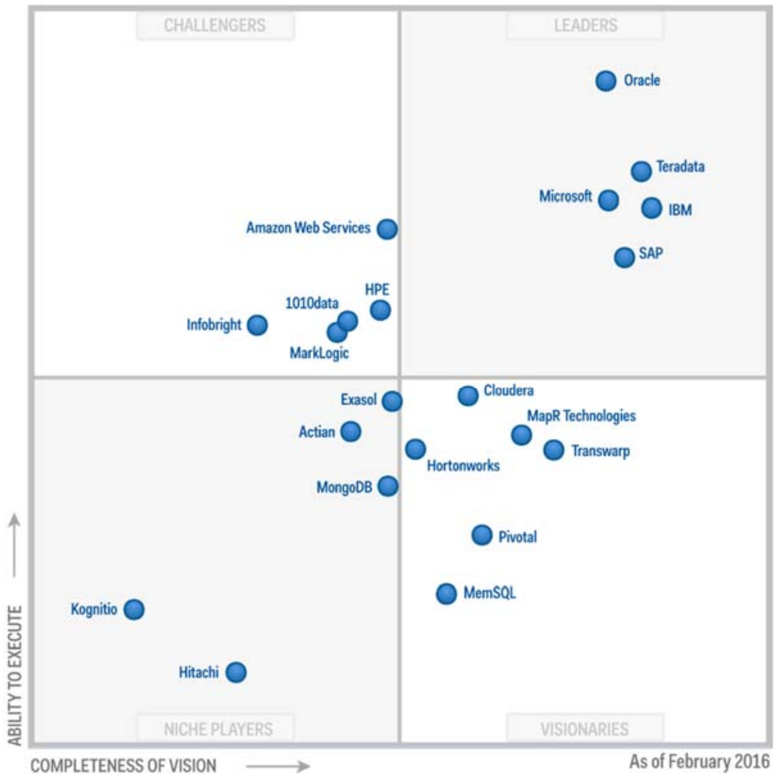
In order to automate the data harvesting, we created a program written in C#, using Microsoft Visual Studio. What began as 200 lines of code was paired down to just 20. Below is the actual code. This program sends a request to Weather Underground's API, retrieves the JSON document from the website, and inserts the data into MongoDB. Microsoft's Visual Studio does not come with inbuilt configuration to connect with a database like MongoDB, unlike SQL, and hence additional packages must be installed, such as the ones below.

```
using MongoDB.Bson;
using MongoDB.Driver;
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Net;
using System.Text;
using System.Threading.Tasks;

namespace BIproject
{
   class Program
   {
      static void Main(string[] args)
      {
         getData();
      }
      public static void getData()
      {
         Console.WriteLine("Getting Data");
         //Declare Url for Api retrieval
         Uri uri = new
Uri(@"http://api.wunderground.com/api/88f2e9a1a85ff5d4/conditions/q/CA/San_Francis
co.json");
         WebRequest webRequest = WebRequest.Create(uri); //creates a web request
         WebResponse response = webRequest.GetResponse(); //declare response object
         StreamReader streamReader = new
StreamReader(response.GetResponseStream()); //reader object
         string responseData = streamReader.ReadToEnd(); //reads data from URL and
stores in local variable
```

```
            Console.WriteLine("Data Received");
            Console.WriteLine(responseData); //writes the string to console

            Console.WriteLine("trying to create client...");

            //initialize the connection to mongoDB server
            var myMongodbClient = new MongoClient();

            //Gets a mongoDB database instance representing a database on this server
            var myMongodbDatabase = myMongodbClient.GetDatabase("Weather_Test");

            //Gets a collection from the database
            var myMongodbCollection =
myMongodbDatabase.GetCollection<BsonDocument>("San_Francisco_Test");

            //Passes the string response into a BSON document.
            BsonDocument latestWeather = BsonDocument.Parse(responseData);

            //Inserts the document into the collection
            myMongodbCollection.InsertOne(latestWeather);

            Console.WriteLine("Inserted Document into MongoDB!");

        }
    }
}
```

Why MongoDB

MongoDB an open source NoSQL, document database management system. It is offered as a cloud service or for local installation. It has been embraced by application developers and system architects. It's known for its ease of application integration and for its strength in processing operational data. MongoDB was added to Gartner's Magic Quadrant for Data Warehouse and Data Management Solutions for Analytics in 2016. It is listed in the Niche Players category.



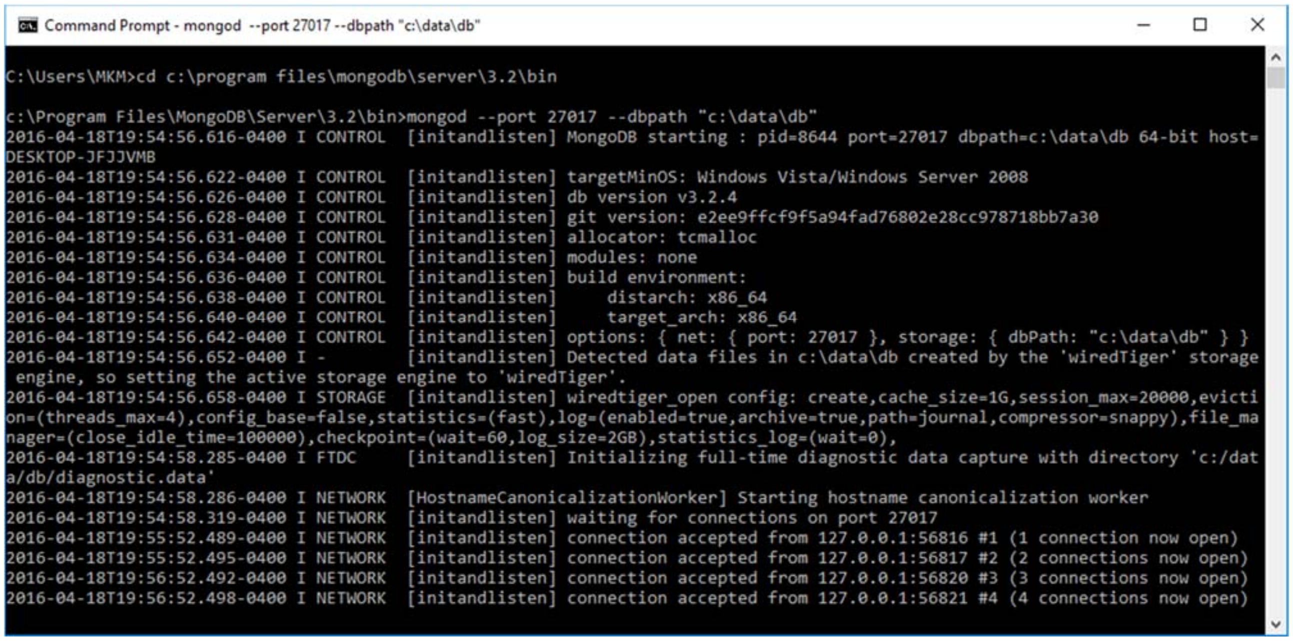
MongoDB's core application is written in C++. Its architecture stores data in collections. Collections are comprised of BSON documents, which is a superset of JSON. MongoDB allows the sharding of collections across multiple machines. Indexes can be created for collections, documents, or data within a document including: fields, compound fields, geospatial data, text (strings). It stores indexes in RAM when possible. The numerous methods of indexing, and the storing of indexes in RAM make data processing and querying very fast.

MongoDB History

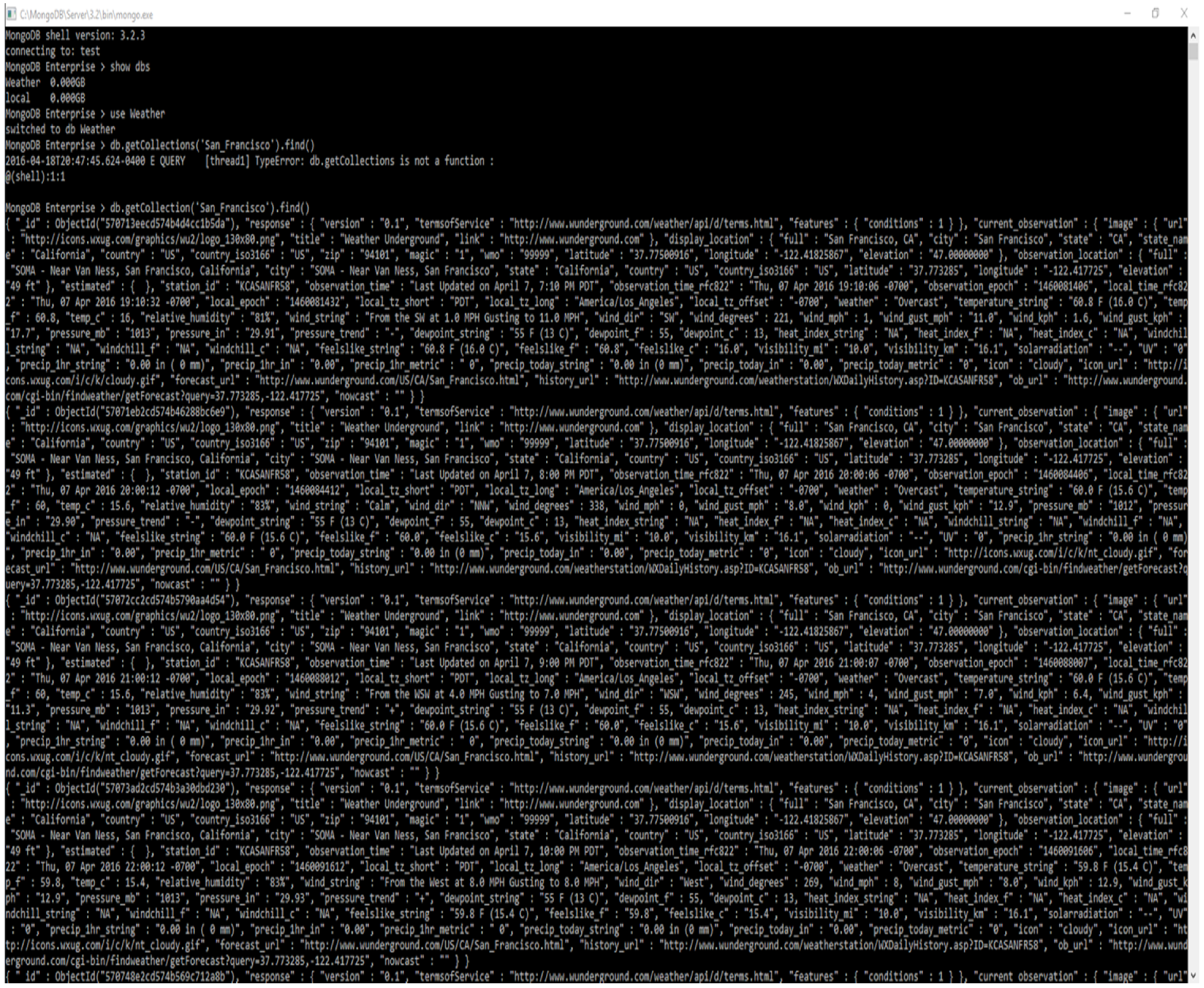
MongoDB, Inc. is the company behind MongoDB. It was founded in 2007, with the intent on becoming a software as service company. It is now open source, which of course means that it is available for free. MongoDB, Inc. offers support commercially. They have thousands of customers using MongoDB and their subscription based support.

Our MongoDB Experience

While it was relatively easy to set-up, MongoDB has no graphical user interface. It opens in Windows using the Command Prompt. Queries, data updates, and administrative operations are performed in MongoDB using the mongo shell. Mongo shell is an interactive JavaScript interface. Since our team does not yet have expertise in JavaScript, we sought a third-party graphical user interface.



```
C:\Users\MKM>cd c:\program files\mongodb\server\3.2\bin

c:\Program Files\MongoDB\Server\3.2\bin>mongod --port 27017 --dbpath "c:\data\db"
2016-04-18T19:54:56.616-0400 I CONTROL  [initandlisten] MongoDB starting : pid=8644 port=27017 dbpath=c:\data\db 64-bit host=
DESKTOP-JFJJVMB
2016-04-18T19:54:56.622-0400 I CONTROL  [initandlisten] targetMinOS: Windows Vista/Windows Server 2008
2016-04-18T19:54:56.626-0400 I CONTROL  [initandlisten] db version v3.2.4
2016-04-18T19:54:56.628-0400 I CONTROL  [initandlisten] git version: e2ee9ffcf9f5a94fad76802e28cc978718bb7a30
2016-04-18T19:54:56.631-0400 I CONTROL  [initandlisten] allocator: tcmalloc
2016-04-18T19:54:56.634-0400 I CONTROL  [initandlisten] modules: none
2016-04-18T19:54:56.636-0400 I CONTROL  [initandlisten] build environment:
2016-04-18T19:54:56.638-0400 I CONTROL  [initandlisten]     distarch: x86_64
2016-04-18T19:54:56.640-0400 I CONTROL  [initandlisten]     target_arch: x86_64
2016-04-18T19:54:56.642-0400 I CONTROL  [initandlisten] options: { net: { port: 27017 }, storage: { dbPath: "c:\data\db" } }
2016-04-18T19:54:56.652-0400 I -        [initandlisten] Detected data files in c:\data\db created by the 'wiredTiger' storage
 engine, so setting the active storage engine to 'wiredTiger'.
2016-04-18T19:54:56.658-0400 I STORAGE  [initandlisten] wiredtiger_open config: create,cache_size=1G,session_max=20000,evicti
on=(threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal,compressor=snappy),file_ma
nager=(close_idle_time=100000),checkpoint=(wait=60,log_size=2GB),statistics_log=(wait=0),
2016-04-18T19:54:58.285-0400 I FTDC     [initandlisten] Initializing full-time diagnostic data capture with directory 'c:/dat
a/db/diagnostic.data'
2016-04-18T19:54:58.286-0400 I NETWORK  [HostnameCanonicalizationWorker] Starting hostname canonicalization worker
2016-04-18T19:54:58.319-0400 I NETWORK  [initandlisten] waiting for connections on port 27017
2016-04-18T19:55:52.489-0400 I NETWORK  [initandlisten] connection accepted from 127.0.0.1:56816 #1 (1 connection now open)
2016-04-18T19:55:52.495-0400 I NETWORK  [initandlisten] connection accepted from 127.0.0.1:56817 #2 (2 connections now open)
2016-04-18T19:56:52.492-0400 I NETWORK  [initandlisten] connection accepted from 127.0.0.1:56820 #3 (3 connections now open)
2016-04-18T19:56:52.498-0400 I NETWORK  [initandlisten] connection accepted from 127.0.0.1:56821 #4 (4 connections now open)
```



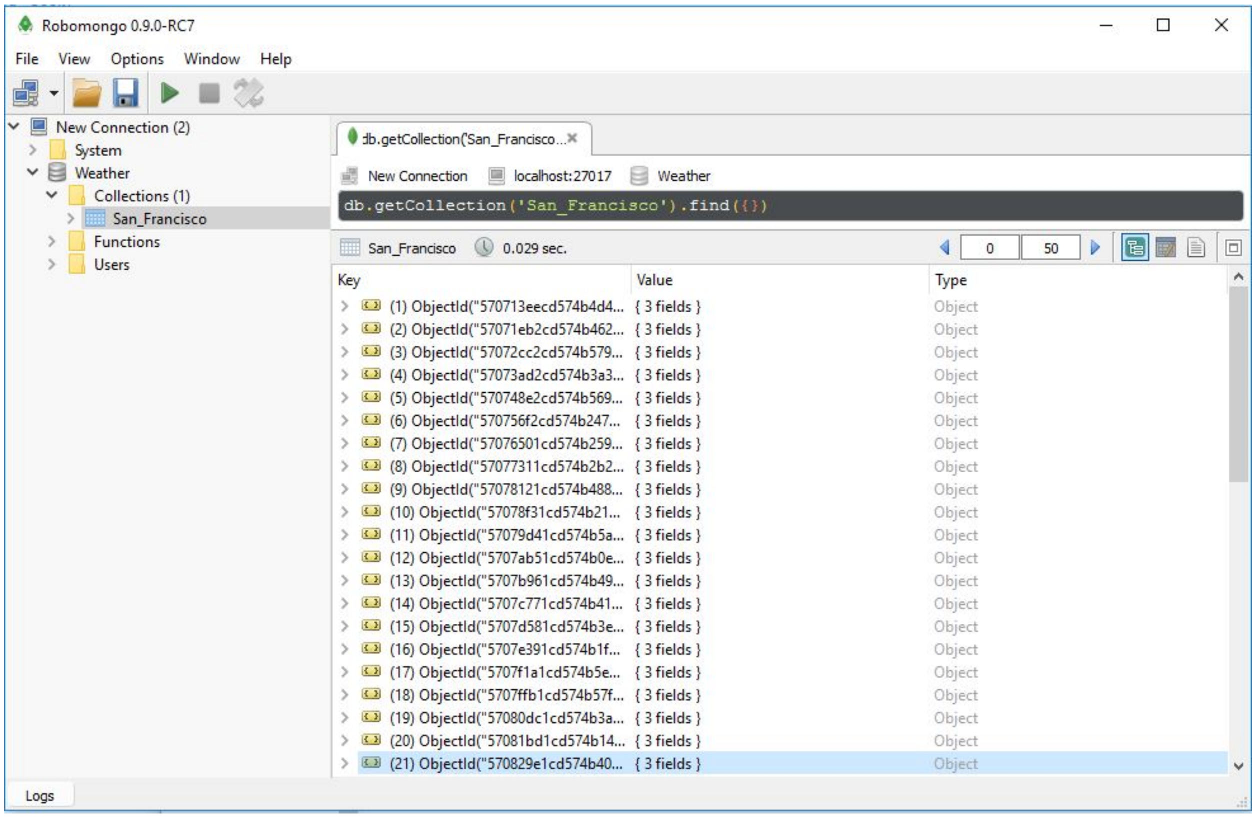
C:\MongoDB\Server\3.2\bin\mongo.exe

```
MongoDB shell version: 3.2.3
connecting to: test
MongoDB Enterprise > show dbs
Weather  0.000GB
local    0.000GB
MongoDB Enterprise > use Weather
switched to db Weather
MongoDB Enterprise > db.getCollections('San_Francisco').find()
2016-04-18T20:47:45.624-0400 E QUERY    [thread1] TypeError: db.getCollections is not a function :
@(shell):1:1

MongoDB Enterprise > db.getCollection('San_Francisco').find()
{ "_id" : ObjectId("570713eecd574b4d4cc1b5da"), "response" : { "version" : "0.1", "termsofService" : "http://www.wunderground.com/weather/api/d/terms.html", "features" : { "conditions" : 1 } }, "current_observation" : { "image" : { "url"
 : "http://icons.wxug.com/graphics/wu2/logo_130x80.png", "title" : "Weather Underground", "link" : "http://www.wunderground.com" }, "display_location" : { "full" : "San Francisco, CA", "city" : "San Francisco", "state" : "CA", "state_nam
e" : "California", "country" : "US", "country_iso3166" : "US", "zip" : "94101", "magic" : "1", "wmo" : "99999", "latitude" : "37.77500916", "longitude" : "-122.41825867", "elevation" : "47.00000000" }, "observation_location" : { "full" :
 "SOMA - Near Van Ness, San Francisco, California", "city" : "SOMA - Near Van Ness, San Francisco", "state" : "California", "country" : "US", "country_iso3166" : "US", "latitude" : "37.773285", "longitude" : "-122.417725", "elevation" :
"49 ft" }, "estimated" : {  }, "station_id" : "KCASANFR58", "observation_time" : "Last Updated on April 7, 7:10 PM PDT", "observation_time_rfc822" : "Thu, 07 Apr 2016 19:10:06 -0700", "observation_epoch" : "1460081406", "local_time_rfc82
2" : "Thu, 07 Apr 2016 19:10:32 -0700", "local_epoch" : "1460081432", "local_tz_short" : "PDT", "local_tz_long" : "America/Los_Angeles", "local_tz_offset" : "-0700", "weather" : "Overcast", "temperature_string" : "60.8 F (16.0 C)", "temp
_f" : 60.8, "temp_c" : 16, "relative_humidity" : "81%", "wind_string" : "From the SW at 1.0 MPH Gusting to 11.0 MPH", "wind_dir" : "SW", "wind_degrees" : 221, "wind_mph" : 1, "wind_gust_mph" : "11.0", "wind_kph" : 1.6, "wind_gust_kph" :
"17.7", "pressure_mb" : "1013", "pressure_in" : "29.91", "pressure_trend" : "-", "dewpoint_string" : "55 F (13 C)", "dewpoint_f" : 55, "dewpoint_c" : 13, "heat_index_string" : "NA", "heat_index_f" : "NA", "heat_index_c" : "NA", "windchil
l_string" : "NA", "windchill_f" : "NA", "windchill_c" : "NA", "feelslike_string" : "60.8 F (16.0 C)", "feelslike_f" : "60.8", "feelslike_c" : "16.0", "visibility_mi" : "10.0", "visibility_km" : "16.1", "solarradiation" : "--", "UV" : "0"
, "precip_1hr_string" : "0.00 in ( 0 mm)", "precip_1hr_in" : "0.00", "precip_1hr_metric" : " 0", "precip_today_string" : "0.00 in (0 mm)", "precip_today_in" : "0.00", "precip_today_metric" : "0", "icon" : "cloudy", "icon_url" : "http://i
cons.wxug.com/i/c/k/cloudy.gif", "forecast_url" : "http://www.wunderground.com/US/CA/San_Francisco.html", "history_url" : "http://www.wunderground.com/weatherstation/WXDailyHistory.asp?ID=KCASANFR58", "ob_url" : "http://www.wunderground.
com/cgi-bin/findweather/getForecast?query=37.773285,-122.417725", "nowcast" : "" } }
{ "_id" : ObjectId("57071eb2cd574b46288bc6e9"), "response" : { "version" : "0.1", "termsofService" : "http://www.wunderground.com/weather/api/d/terms.html", "features" : { "conditions" : 1 } }, "current_observation" : { "image" : { "url"
 : "http://icons.wxug.com/graphics/wu2/logo_130x80.png", "title" : "Weather Underground", "link" : "http://www.wunderground.com" }, "display_location" : { "full" : "San Francisco, CA", "city" : "San Francisco", "state" : "CA", "state_nam
e" : "California", "country" : "US", "country_iso3166" : "US", "zip" : "94101", "magic" : "1", "wmo" : "99999", "latitude" : "37.77500916", "longitude" : "-122.41825867", "elevation" : "47.00000000" }, "observation_location" : { "full" :
 "SOMA - Near Van Ness, San Francisco, California", "city" : "SOMA - Near Van Ness, San Francisco", "state" : "California", "country" : "US", "country_iso3166" : "US", "latitude" : "37.773285", "longitude" : "-122.417725", "elevation" :
"49 ft" }, "estimated" : {  }, "station_id" : "KCASANFR58", "observation_time" : "Last Updated on April 7, 8:00 PM PDT", "observation_time_rfc822" : "Thu, 07 Apr 2016 20:00:06 -0700", "observation_epoch" : "1460084406", "local_time_rfc82
2" : "Thu, 07 Apr 2016 20:00:12 -0700", "local_epoch" : "1460084412", "local_tz_short" : "PDT", "local_tz_long" : "America/Los_Angeles", "local_tz_offset" : "-0700", "weather" : "Overcast", "temperature_string" : "60.0 F (15.6 C)", "temp
_f" : 60, "temp_c" : 15.6, "relative_humidity" : "83%", "wind_string" : "Calm", "wind_dir" : "NNW", "wind_degrees" : 338, "wind_mph" : 0, "wind_gust_mph" : "8.0", "wind_kph" : 0, "wind_gust_kph" : "12.9", "pressure_mb" : "1012", "pressur
e_in" : "29.90", "pressure_trend" : "-", "dewpoint_string" : "55 F (13 C)", "dewpoint_f" : 55, "dewpoint_c" : 13, "heat_index_string" : "NA", "heat_index_f" : "NA", "heat_index_c" : "NA", "windchill_string" : "NA", "windchill_f" : "NA",
"windchill_c" : "NA", "feelslike_string" : "60.0 F (15.6 C)", "feelslike_f" : "60.0", "feelslike_c" : "15.6", "visibility_mi" : "10.0", "visibility_km" : "16.1", "solarradiation" : "--", "UV" : "0", "precip_1hr_string" : "0.00 in ( 0 mm)
", "precip_1hr_in" : "0.00", "precip_1hr_metric" : " 0", "precip_today_string" : "0.00 in (0 mm)", "precip_today_in" : "0.00", "precip_today_metric" : "0", "icon" : "cloudy", "icon_url" : "http://icons.wxug.com/i/c/k/nt_cloudy.gif", "for
ecast_url" : "http://www.wunderground.com/US/CA/San_Francisco.html", "history_url" : "http://www.wunderground.com/weatherstation/WXDailyHistory.asp?ID=KCASANFR58", "ob_url" : "http://www.wunderground.com/cgi-bin/findweather/getForecast?q
uery=37.773285,-122.417725", "nowcast" : "" } }
{ "_id" : ObjectId("57072cc2cd574b5790aa4d54"), "response" : { "version" : "0.1", "termsofService" : "http://www.wunderground.com/weather/api/d/terms.html", "features" : { "conditions" : 1 } }, "current_observation" : { "image" : { "url"
 : "http://icons.wxug.com/graphics/wu2/logo_130x80.png", "title" : "Weather Underground", "link" : "http://www.wunderground.com" }, "display_location" : { "full" : "San Francisco, CA", "city" : "San Francisco", "state" : "CA", "state_nam
e" : "California", "country" : "US", "country_iso3166" : "US", "zip" : "94101", "magic" : "1", "wmo" : "99999", "latitude" : "37.77500916", "longitude" : "-122.41825867", "elevation" : "47.00000000" }, "observation_location" : { "full" :
 "SOMA - Near Van Ness, San Francisco, California", "city" : "SOMA - Near Van Ness, San Francisco", "state" : "California", "country" : "US", "country_iso3166" : "US", "latitude" : "37.773285", "longitude" : "-122.417725", "elevation" :
"49 ft" }, "estimated" : {  }, "station_id" : "KCASANFR58", "observation_time" : "Last Updated on April 7, 9:00 PM PDT", "observation_time_rfc822" : "Thu, 07 Apr 2016 21:00:07 -0700", "observation_epoch" : "1460088007", "local_time_rfc82
2" : "Thu, 07 Apr 2016 21:00:12 -0700", "local_epoch" : "1460088012", "local_tz_short" : "PDT", "local_tz_long" : "America/Los_Angeles", "local_tz_offset" : "-0700", "weather" : "Overcast", "temperature_string" : "60.0 F (15.6 C)", "temp
_f" : 60, "temp_c" : 15.6, "relative_humidity" : "83%", "wind_string" : "From the WSW at 4.0 MPH Gusting to 7.0 MPH", "wind_dir" : "WSW", "wind_degrees" : 245, "wind_mph" : 4, "wind_gust_mph" : "7.0", "wind_kph" : 6.4, "wind_gust_kph" :
"11.3", "pressure_mb" : "1013", "pressure_in" : "29.92", "pressure_trend" : "+", "dewpoint_string" : "55 F (13 C)", "dewpoint_f" : 55, "dewpoint_c" : 13, "heat_index_string" : "NA", "heat_index_f" : "NA", "heat_index_c" : "NA", "windchil
l_string" : "NA", "windchill_f" : "NA", "windchill_c" : "NA", "feelslike_string" : "60.0 F (15.6 C)", "feelslike_f" : "60.0", "feelslike_c" : "15.6", "visibility_mi" : "10.0", "visibility_km" : "16.1", "solarradiation" : "--", "UV" : "0"
, "precip_1hr_string" : "0.00 in ( 0 mm)", "precip_1hr_in" : "0.00", "precip_1hr_metric" : " 0", "precip_today_string" : "0.00 in (0 mm)", "precip_today_in" : "0.00", "precip_today_metric" : "0", "icon" : "cloudy", "icon_url" : "http://i
cons.wxug.com/i/c/k/nt_cloudy.gif", "forecast_url" : "http://www.wunderground.com/US/CA/San_Francisco.html", "history_url" : "http://www.wunderground.com/weatherstation/WXDailyHistory.asp?ID=KCASANFR58", "ob_url" : "http://www.undergrou
nd.com/cgi-bin/findweather/getForecast?query=37.773285,-122.417725", "nowcast" : "" } }
{ "_id" : ObjectId("57073ad2cd574b3a30dbd230"), "response" : { "version" : "0.1", "termsofService" : "http://www.wunderground.com/weather/api/d/terms.html", "features" : { "conditions" : 1 } }, "current_observation" : { "image" : { "url"
 : "http://icons.wxug.com/graphics/wu2/logo_130x80.png", "title" : "Weather Underground", "link" : "http://www.wunderground.com" }, "display_location" : { "full" : "San Francisco, CA", "city" : "San Francisco", "state" : "CA", "state_nam
e" : "California", "country" : "US", "country_iso3166" : "US", "zip" : "94101", "magic" : "1", "wmo" : "99999", "latitude" : "37.77500916", "longitude" : "-122.41825867", "elevation" : "47.00000000" }, "observation_location" : { "full" :
 "SOMA - Near Van Ness, San Francisco, California", "city" : "SOMA - Near Van Ness, San Francisco", "state" : "California", "country" : "US", "country_iso3166" : "US", "latitude" : "37.773285", "longitude" : "-122.417725", "elevation" :
"49 ft" }, "estimated" : {  }, "station_id" : "KCASANFR58", "observation_time" : "Last Updated on April 7, 10:00 PM PDT", "observation_time_rfc822" : "Thu, 07 Apr 2016 22:00:06 -0700", "observation_epoch" : "1460091606", "local_time_rfc8
22" : "Thu, 07 Apr 2016 22:00:12 -0700", "local_epoch" : "1460091612", "local_tz_short" : "PDT", "local_tz_long" : "America/Los_Angeles", "local_tz_offset" : "-0700", "weather" : "Overcast", "temperature_string" : "59.8 F (15.4 C)", "tem
p_f" : 59.8, "temp_c" : 15.4, "relative_humidity" : "83%", "wind_string" : "From the West at 8.0 MPH Gusting to 8.0 MPH", "wind_dir" : "West", "wind_degrees" : 269, "wind_mph" : 8, "wind_gust_mph" : "8.0", "wind_kph" : 12.9, "wind_gust_k
ph" : "12.9", "pressure_mb" : "1013", "pressure_in" : "29.93", "pressure_trend" : "+", "dewpoint_string" : "55 F (13 C)", "dewpoint_f" : 55, "dewpoint_c" : 13, "heat_index_string" : "NA", "heat_index_f" : "NA", "heat_index_c" : "NA", "wi
ndchill_string" : "NA", "windchill_f" : "NA", "windchill_c" : "NA", "feelslike_string" : "59.8 F (15.4 C)", "feelslike_f" : "59.8", "feelslike_c" : "15.4", "visibility_mi" : "10.0", "visibility_km" : "16.1", "solarradiation" : "--", "UV"
 : "0", "precip_1hr_string" : "0.00 in ( 0 mm)", "precip_1hr_in" : "0.00", "precip_1hr_metric" : " 0", "precip_today_string" : "0.00 in (0 mm)", "precip_today_in" : "0.00", "precip_today_metric" : "0", "icon" : "cloudy", "icon_url" : "ht
tp://icons.wxug.com/i/c/k/nt_cloudy.gif", "forecast_url" : "http://www.wunderground.com/US/CA/San_Francisco.html", "history_url" : "http://www.wunderground.com/weatherstation/WXDailyHistory.asp?ID=KCASANFR58", "ob_url" : "http://www.wund
erground.com/cgi-bin/findweather/getForecast?query=37.773285,-122.417725", "nowcast" : "" } }
{ "_id" : ObjectId("570748e2cd574b569c712a8b"), "response" : { "version" : "0.1", "termsofService" : "http://www.wunderground.com/weather/api/d/terms.html", "features" : { "conditions" : 1 } }, "current_observation" : { "image" : { "url"
```

Robomongo

There are many third-party graphical user interfaces (GUI’s) that integrate with MongoDB. Robomongo is an open source desktop application GUI that embeds the mongo shell and provides identical functionality. Because of the simplicity of our database, we did not actually use Robomongo extensively.

Early in the harvesting of our data, we reduced the frequency of our API requests to Weather Underground from every 15 minutes to every hour. For consistency, we wanted to eliminate the 3 additional JSON documents per hour from our database. Robomongo made it easy to identify the extra data and eliminate it from our database. Of course, we would have kept that data and queried only the data that we wanted, if the scope of this project were different. For our needs, the additional data was not relevant and could be eliminated without repercussion.

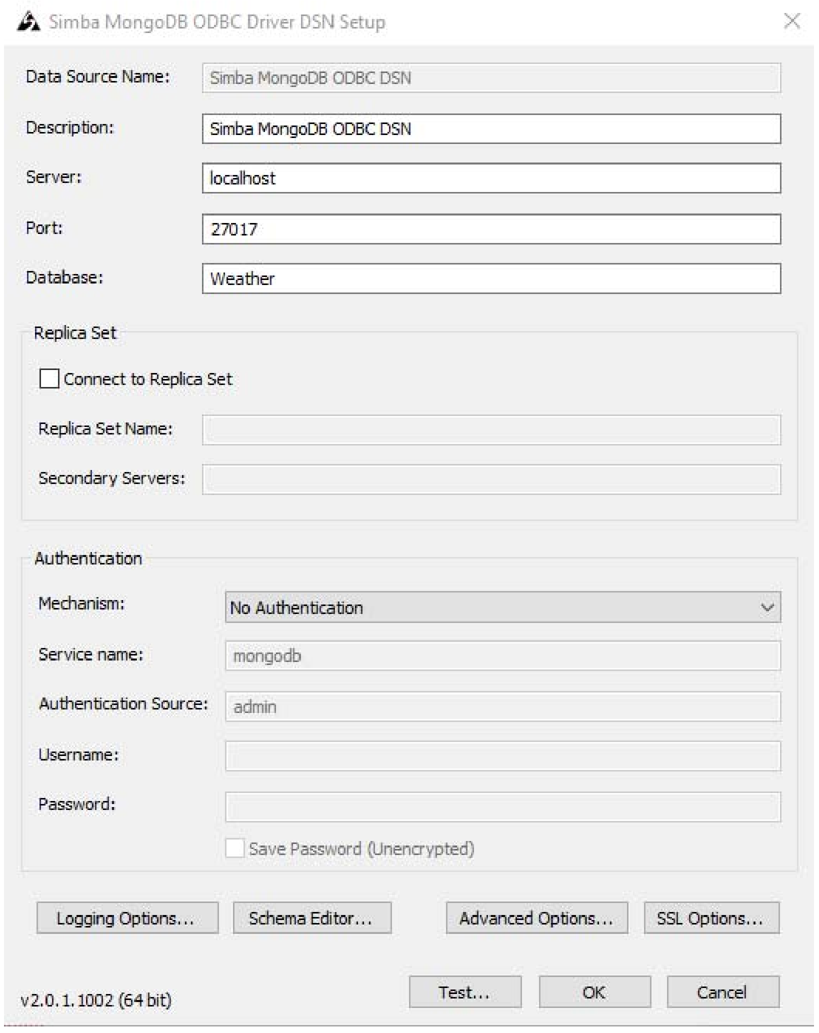


```
db.getCollection('San_Francisco').find({})
```

Key	Value	Type
(1) ObjectId("570713eecd574b4d4...	{ 3 fields }	Object
(2) ObjectId("57071eb2cd574b462...	{ 3 fields }	Object
(3) ObjectId("57072cc2cd574b579...	{ 3 fields }	Object
(4) ObjectId("57073ad2cd574b3a3...	{ 3 fields }	Object
(5) ObjectId("570748e2cd574b569...	{ 3 fields }	Object
(6) ObjectId("570756f2cd574b247...	{ 3 fields }	Object
(7) ObjectId("57076501cd574b259...	{ 3 fields }	Object
(8) ObjectId("57077311cd574b2b2...	{ 3 fields }	Object
(9) ObjectId("57078121cd574b488...	{ 3 fields }	Object
(10) ObjectId("57078f31cd574b21...	{ 3 fields }	Object
(11) ObjectId("57079d41cd574b5a...	{ 3 fields }	Object
(12) ObjectId("5707ab51cd574b0e...	{ 3 fields }	Object
(13) ObjectId("5707b961cd574b49...	{ 3 fields }	Object
(14) ObjectId("5707c771cd574b41...	{ 3 fields }	Object
(15) ObjectId("5707d581cd574b3e...	{ 3 fields }	Object
(16) ObjectId("5707e391cd574b1f...	{ 3 fields }	Object
(17) ObjectId("5707f1a1cd574b5e...	{ 3 fields }	Object
(18) ObjectId("5707ffb1cd574b57f...	{ 3 fields }	Object
(19) ObjectId("57080dc1cd574b3a...	{ 3 fields }	Object
(20) ObjectId("57081bd1cd574b14...	{ 3 fields }	Object
(21) ObjectId("570829e1cd574b40...	{ 3 fields }	Object

Simba Driver for MongoDB with SQL Connector

Simba Technologies offers an Open Database Connectivity (ODBC) driver that brilliantly connects MongoDB to Tableau. Though using the latest versions of both MongoDB and Tableau, it is possible to integrate the two platforms directly, the process is complex and there is opportunity for data corruption. Configuring the Simba driver was relatively easy.


Simba MongoDB ODBC Driver DSN Setup

Data Source Name:	Simba MongoDB ODBC DSN
Description:	Simba MongoDB ODBC DSN
Server:	localhost
Port:	27017
Database:	Weather

Replica Set

☐ Connect to Replica Set

Replica Set Name:	
Secondary Servers:	

Authentication

Mechanism:	No Authentication
Service name:	mongodb
Authentication Source:	admin
Username:	
Password:	

☐ Save Password (Unencrypted)

v2.0.1.1002 (64 bit)

Why Tableau

Tableau is a platform that help users analyze raw data and transform it into business intelligence, without requiring coding knowledge. More specifically, it allows users to create visualizations of data, which helps senior managers make better decisions for their organizations, backed by their data. Gartner's Magic Quadrant for Business Intelligence and Analytics Platforms ranks Tableau in the Leaders quadrant, a distinction given only to two other software companies. It ranked highest in of all companies evaluated in Ability to Execute.

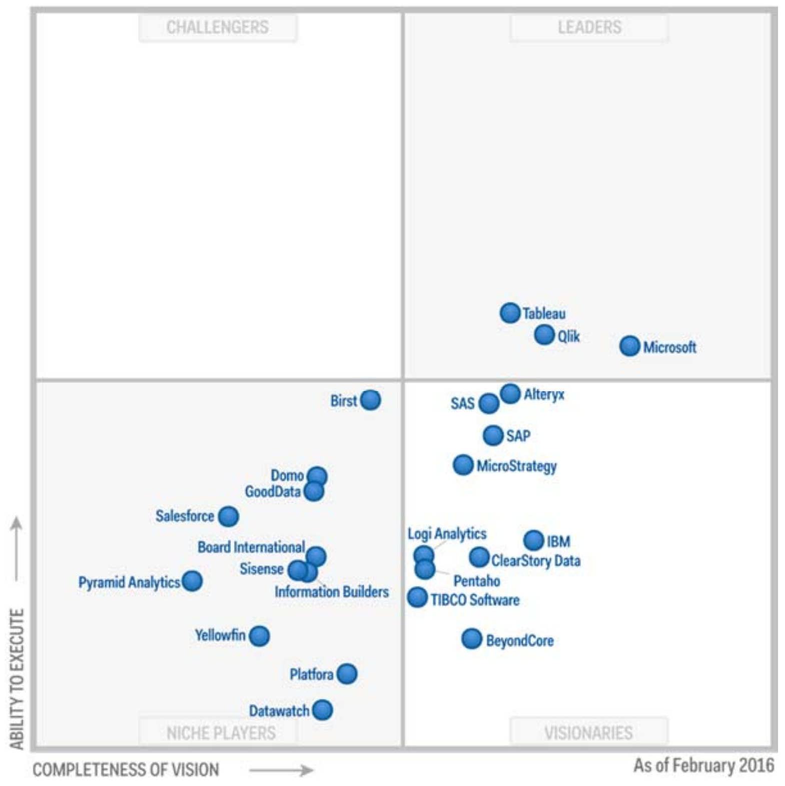


Tableau History

Tableau was established in 2003. Its founders, Chis Stolte, Christian Chabot, and Pat Hanrahan were connected through Stanford University. Chabot is the CEO, and earned a BS in Engineering and an MBA from Stanford. Chris Stolte is the CDO and earned his Ph.D. in Computer Science from Stanford. Pat Hanrahan is the Chief Scientist at Tableau and is also the CANON Professor of Computer Science and Electrical engineering at Stanford. Hanrahan has the added distinction of being a founding employee at Pixar, and has won three Academy Awards for his computer graphics production.

Tableau is a publicly owned company. It is traded on the New York Stock Exchange, as "DATA". Tableau's initial public offering launched May 17, 2013. It provided the following prospectus summary to the U.S. Securities and Exchange Commission, for their IPO consideration:

> ***Company Overview***
>
> *Our mission is to help people see and understand data.*
>
> *Our software products put the power of data into the hands of everyday people, allowing a broad population of business users to engage with their data, ask questions, solve problems and create value.*
>
> *Based on innovative core technologies originally developed at Stanford University, our products dramatically reduce the complexity, inflexibility and expense associated with traditional business intelligence applications. We aim to make our products easy to use, ubiquitous and as deeply-rooted in the workplace as spreadsheets are today.*
>
> *Our software is designed for anyone with data and questions. We are democratizing the use of business analytics software by allowing people to access information, perform analysis and share results without assistance from technical specialists. By putting powerful, self-service analytical technology directly into the hands of people who make decisions with data, we seek to accelerate the pace of informed and intelligent decision making. This enables our customers to create better workplaces, with happier employees who are empowered to more fully express their ingenuity and creativity.*
>
> *Our products are used by people of diverse skill levels across all kinds of organizations, including Fortune 500 corporations, small and medium-sized businesses, government agencies, universities, research institutions and non-profits. Organizations employ our products in a broad range of use cases such as increasing sales, streamlining operations, improving customer service, managing investments, assessing quality and safety, studying and treating diseases, completing academic research, addressing environmental problems and improving education. Our products are flexible and capable enough to help a single user on a laptop analyze data from a simple spreadsheet, or to enable thousands of users across an enterprise to execute complex queries against massive databases.*

Pricing and Specifications

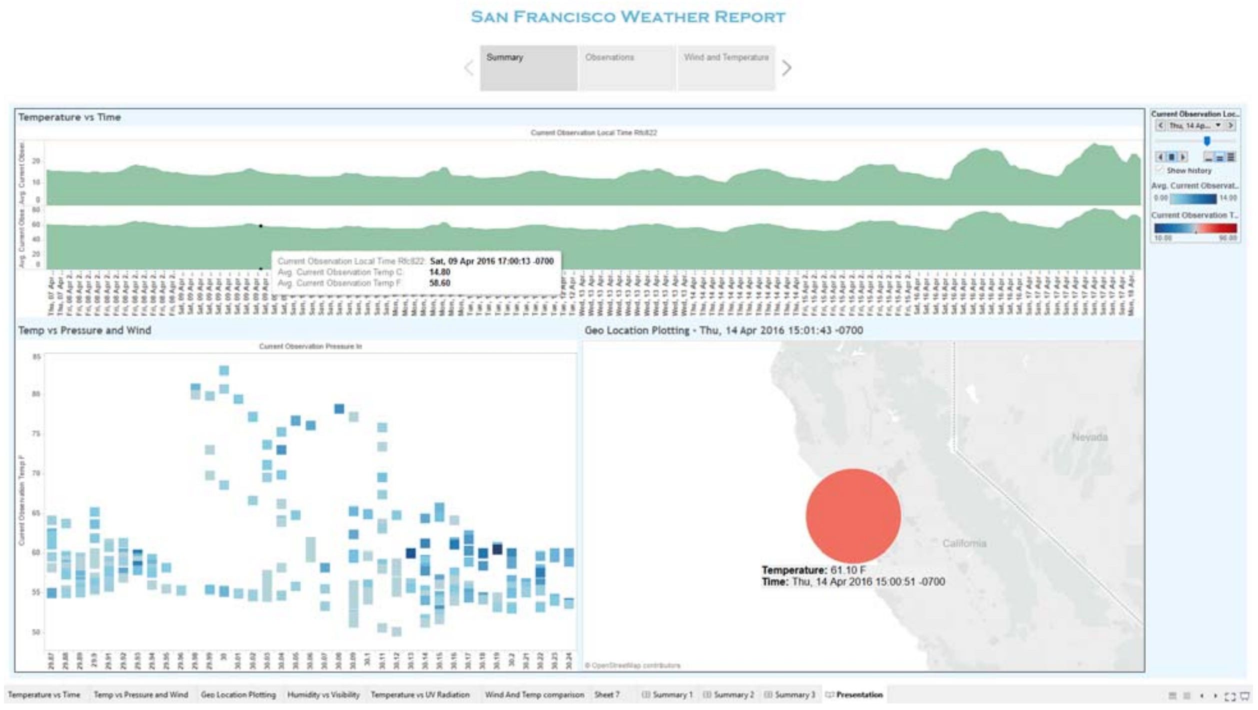
Tableau is offered in two platforms. The Personal Edition sells for $999. The Professional Edition Sells for $1,999. Both editions include 1 year of updates and support.

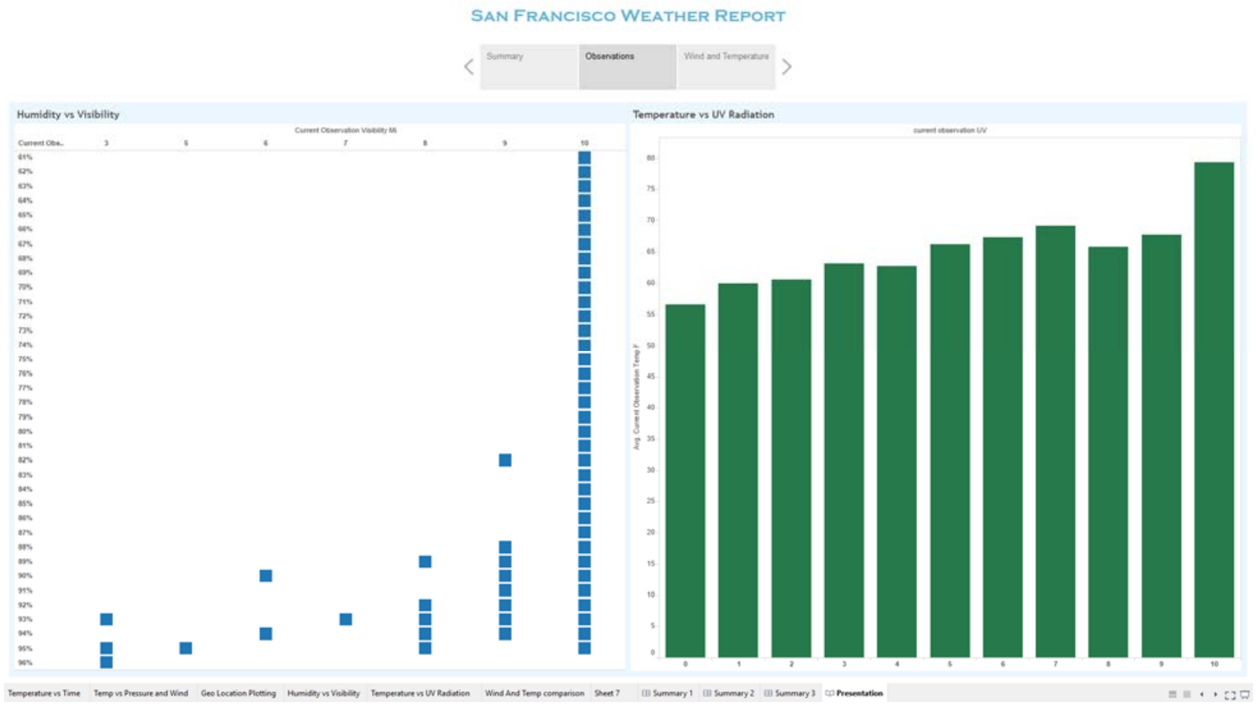
Tableau Personal Edition can only connect to a limited number of data sources, including: Microsoft Excel spreadsheets, text files, Microsoft Access databases, OData, Azure Marketplace DataMarket, and Tableau Data Extract. Tableau Professional Edition can connect to 39 standard data sources, any databases that are ODBC 3.0 compliant, and a growing number of data sources through their Web Data Connector. Data from any website that publishes data in JSON, SML, or HTML formats, Tableau can access via their Web Data Connector. Using Tableau's Software Development Kit (SDK), users can create connections to web content using JavaScript and HTML.

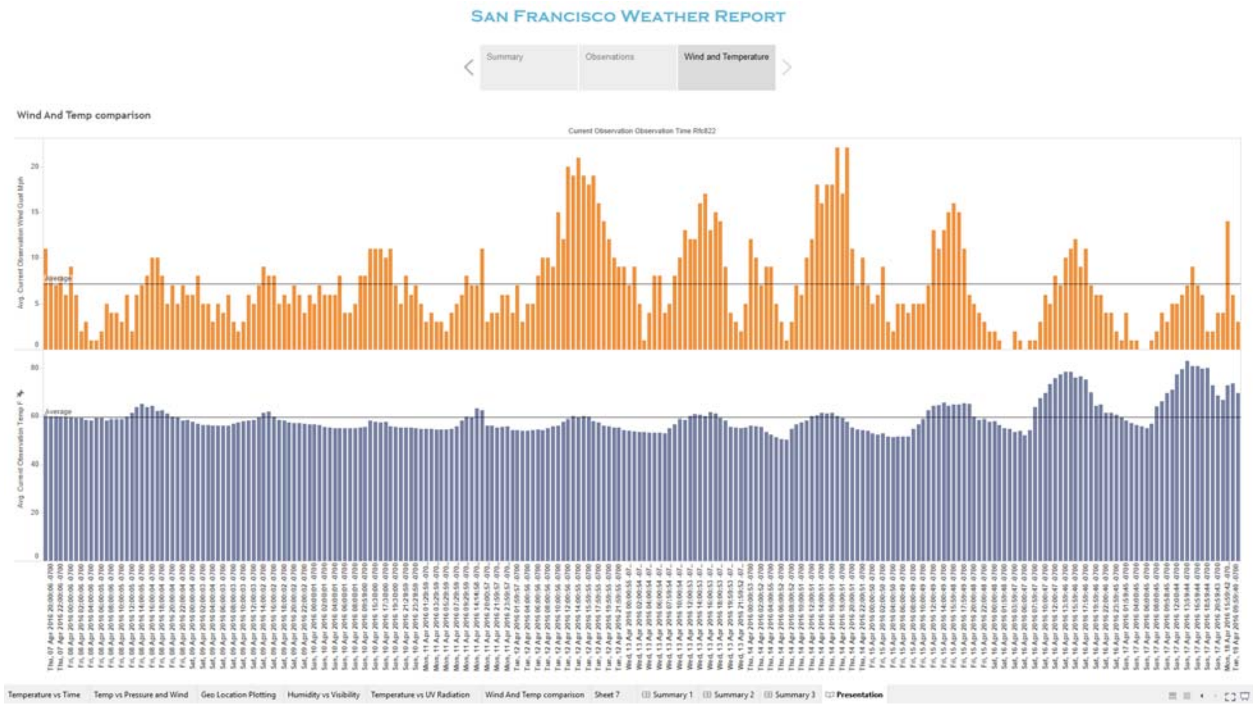
Our Tableau Experience

Once Tableau was connected to our database, it was easy to begin creating visualizations with our data. Tableau's basic function are intuitive and the features are familiar those of other common business applications. Within Tableau, data can be visualized and analyzed in "sheets", a name shared with Microsoft Excel. Sheets can be combined to create "dashboards". Sheets and dashboards can be added to "stories", which are used for presentation.

Within the application, dimensions and measures are listed. The data can be converted from measures to dimension, discrete to continuous, string to number or date; and vise versa for all. The data properties can be changed too. You can drag and drop them into rows and columns to begin creating visualizations. Once placed into a visualization, the data can be sorted. Calculations can be made from the data, including: running total, difference, percent difference, percent of total, rank, percentile, moving average, YTD total, compound growth rate, year over year growth, and YTD growth. Analysis can be added to the visualization including: constant line, average line, median with quartiles, box plot, totals, average with 95% confidence level, median with 95% confidence level, trend line, and forecast.


San Francisco Weather Report
Summary
Observations
Wind and Temperature
Temperature vs Time
Current Observation Local Time Rfc822
Current Observation Local Time Rfc822: Sat, 09 Apr 2016 17:00:13 -0700
Avg. Current Observation Temp C: 14.80
Avg. Current Observation Temp F: 58.60
Temp vs Pressure and Wind
Current Observation Pressure In
Current Observation Temp F
Geo Location Plotting - Thu, 14 Apr 2016 15:01:43 -0700
Nevada
California
Temperature: 61.10 F
Time: Thu, 14 Apr 2016 15:00:51 -0700
© OpenStreetMap contributors
Current Observation Loc...
Show history
Avg. Current Observat...
Current Observation T...
Temperature vs Time
Temp vs Pressure and Wind
Geo Location Plotting
Humidity vs Visibility
Temperature vs UV Radiation
Wind And Temp comparison
Sheet 7
Summary 1
Summary 2
Summary 3
Presentation


SAN FRANCISCO WEATHER REPORT
Summary
Observations
Wind and Temperature
Humidity vs Visibility
Current Observation Visibility Mi
Temperature vs UV Radiation
current observation UV
Avg. Current Observation Temp F
Temperature vs Time
Temp vs Pressure and Wind
Geo Location Plotting
Humidity vs Visibility
Temperature vs UV Radiation
Wind And Temp comparison
Sheet 7
Summary 1
Summary 2
Summary 3
Presentation


SAN FRANCISCO WEATHER REPORT
Summary
Observations
Wind and Temperature
Wind And Temp comparison
Current Observation Observation Time Rfc822
Avg. Current Observation Wind Gust Mph
Average
Avg. Current Observation Temp F
Average
Temperature vs Time
Temp vs Pressure and Wind
Geo Location Plotting
Humidity vs Visibility
Temperature vs UV Radiation
Wind And Temp comparison
Sheet 7
Summary 1
Summary 2
Summary 3
Presentation

Tableau's Competition

There are other data discovery and visualization tools on the market, which are similar to Tableau. Qlik and Tibco Spotfire are comparable. They are designed for data analysts and technically-oriented business users. Their focus is not primarily reporting and monitoring, but rather ad-hoc analysis of multiple data sources. These tools provide data analysts with an intuitive way to sift through large volumes of data, to expose patterns and outliers hidden in the data. They replace the customary rows and columns of traditional data presentations with graphical pictures and charts.

These tools have been successful, largely because of the low cost of implementation and because they do not require IT support. Ease of use is another key feature encouraging adoption. Visualization tools allow end users with some comfort level in data analysis to access multiple diverse data sources and display the results in visually compelling ways.

Why Data Visualization is Important

Technological advances have made data visualization more prevalent and powerful than ever before, increasing the importance of business intelligence. Tableau leads the world in making the data visualization process available to business users of every background and industry. Businesses around the globe realize that the ability to visualize data effectively leads directly to better understanding, insight and better business decisions.

Tableau Software enables businesses to keep pace with the evolving technology landscape and outperform competitors through an adaptive and intuitive means of visualizing their data.

Key features to Tableau include:

- Pre-built visualization formats such as heat maps and scatter plots.
- Location analytics and geographic visualizations.
- Predictive analytics.
- Support for machine-learning models.
- Pattern recognition and data mining.
- Integration with R and other statistical packages.

Organizations can use tools to:

- Access and absorb information in new ways which are more constructive than they have been in the past.
- Visualize patterns and relationships between operational data and business data.
- Act on emerging trends immediately and therefore more effectively.
- Manipulate and interact with data in such a way that's never been possible in the past.
- Create a culture in which everyone, organization-wide, understands the value of data.

Works Cited

"Tableau Software." Tableau.com. Tableau Software Inc. Web. 15 Feb. 2016. <http://www.tableau.com/>.

"Form S-1 Registration Statement." Sec.gov. U.S Securities and Exchange Commision, 2 Apr. 2013. Web. 15 Feb. 2016. <http://www.sec.gov/Archives/edgar/data/1303652/000119312513138700/d469057ds1.htm>.

Parenteau, Sallam, and Howson. "Magic Quadrant for Business Intelligence and Analytics Platforms." Gartner. 04 Feb. 2016. Web. 20 Feb. 2016. <http://www.gartner.com/document/3200317?ref=solrResearch&refval=164792925&qid=56c961ae28e5d08bb968696032cc30f8>.

Edijlali, Roxane, and Mark Beyer. "Magic Quadrant for Data Warehouse and Data Management Solutions for Analytics." Gartner. 25 Feb. 2016. Web. 5 Mar. 2016. <http://www.gartner.com/document/3225818?ref=solrAll&refval=164793516&qid=b9de88dca7a8ae1392a891f1ab0056cd>.

"About Us." MongoDB. Web. 9 Mar. 2016. <https://www.mongodb.com/company>.

"DOCUMENTATION." *API*. Web. 21 Mar. 2016. <https://www.wunderground.com/weather/api/d/docs?MR=1>.
"Robomongo." *Robomongo*. Web. 21 Mar. 2016. <https://robomongo.org/>. Source for the robomongo application.