TIBCO®

TIBCO Spotfire Best Practices for Data Access and Hadoop

Today, massive amounts of data are being created from applications, systems, and IoT devices and sensors. To find and exploit opportunities, reduce risk, and delight customers, you need to analyze this data—and TIBCO Spotfire® streaming analytics meets the needs of both business and technical users across the enterprise. With its interactive dashboards, visualizations, and predictive and event-driven analytics, Spotfire helps users develop insights from virtually all data sets and data volumes.

This paper provides some frequently asked questions with answers about TIBCO Spotfire support of Apache® Hadoop®, Spark™, and big data.

WHAT ARE THE WAYS THAT SPOTFIRE CAN CONSUME DATA?

Spotfire can deliver information and insight from data in three ways. Let's review before diving into best practices.

1. **In-Memory.** The Spotfire in-memory aggregation engine is very powerful and consistently fast across small and large data volumes. With transparent scaling, the in-memory engine handles hundreds of millions of rows of data.
2. **In-Database.** Leave big data where it is and aggregate in the database. Spotfire can push SQL and Multidimensional Expressions (MDX) to massively parallel processing, NoSQL, and other hardware-accelerated databases including multi-dimensional cubes where the aggregation queries are executed.
3. **Data-On-Demand.** For a hybrid mix of in-memory and in-database queries, Spotfire can pull subsets of big data into memory based on user selection in the analysis. For example, the user marks data in a visualization, drilling down into details. The detail data, stored in a big data store such as Hadoop, is retrieved to support the drill down, cached for re-use, and swapped out when no longer needed.

All three methods can be leveraged in any one analysis, across all the data sources required to support the analysis. The result brings the fastest response times to end users while also leveraging investment in high-end data repositories and respecting data governance against all enterprise data sources.

WHEN SHOULD WE LOAD DATA UP-FRONT, AND WHEN SHOULD WE USE ON-DEMAND?

Spotfire can extract data from a data source upfront or load slices of data on-demand based on your interactions with the Spotfire clients. Loading static data upfront has many benefits. You have all data at hand during your analysis session, with no loading times. You can analyze your data off-line from the data source and choose when to refresh data.

Data on demand however, enables you to analyze more data; it's really a big-data feature. It enables more data to be analyzed by providing slices of data from the data source based on your interactions with the analysis. Data on-demand is frequently used for extracting data into the Spotfire in-memory engine, but is also available for in-database data tables. On-demand in combination with in-database reduces the amount of rows processed by in-database queries, thus enabling analytics across larger data sets and speeding up visualization rendering. The following table provides some of the pros and cons of each method.

SCENARIO	LOADING DATA UPFRONT/ STATIC	LOADING DATA ON-DEMAND
DATA SIZE	The amount of data that can be analyzed depends on how much memory the client machine has and how long it's feasible to wait for data to load. For larger data volumes, loading is often scheduled to be performed at night. If this option is feasible, it's the recommended data loading option.	On-demand enables more data to be analyzed by only providing the data needed at a given time. It works for both in-memory and in-database.
LOADING TIMES	All loading time is upfront, before any analytics can be done.	Data from the previous session is loaded immediately because it's cached in the analysis file. New data is loaded during the analytics session as needed, which means there are wait times every now and then during the analysis.

WE ARE REPLACING OUR SQL DATABASES WITH HADOOP. WHAT CONSIDERATIONS ARE THERE FROM A SPOTFIRE PERSPECTIVE?

Just like with SQL databases, Spotfire connects to Hadoop's SQL interfaces. It's important to carefully select a Hadoop query engine to avoid surprises later on. While Spotfire can connect to generic Hadoop through JDBC, a SQL handler such as Apache Hive™, Impala, Spark SQL, or Apache Drill is highly recommended. Each offers different aggregation functions and capabilities. Here are a few things to consider:

CONNECTION METHOD

Spotfire lets you analyze data from data tables. These data tables can be copied into the Spotfire in-memory engine or kept in the external data source. In the latter case, the in-database data tables are only represented by a metadata view in Spotfire.

IN-MEMORY	LIVE QUERIES/ IN-DATABASE	IN-MEMORY + LIVE QUERIES
Queries are executed in the Spotfire in-memory data engine that provides extremely fast calculations on any data set and enables all Spotfire features. On-demand data tables are supported.	Queries are executed in the external data source's data engine. Expect the ability to work with the largest data volumes, but also longer wait times for visualizations and filters to refresh compared to in-memory data tables. On-demand data tables are supported.	The key to success is often to combine data loading methods in the same analysis. For example, load as much as possible into memory upfront. Then use live queries for the master view of big data aggregates. When users drill down into the details, load row level transactions into memory using on-demand.

QUERY SPEED

With higher query speed, more data and more complex calculations can be analyzed using in-database data tables and live queries. Higher query speeds also mean that (scheduled) data extracts take less time to complete.

SQL DATABASE	HADOOP
Queries are executed in the Spotfire in-memory data engine that provides extremely fast calculations on any data set and enables all Spotfire features. On-demand data tables are supported.	With Hadoop, query speeds are often not on par with a SQL database, but on the other hand, it's possible to scale across virtually unlimited amounts of data. Hive on MapReduce, Tez, and Spark are not recommended for in-database and interactive live querying. Expect better performance with the SQL engines, Hive with LLAP, Impala, and Spark SQL. There are also databases tightly integrated with Hadoop. Examples are Apache HAWQ and IBM BigSQL, which are represented as a Greenplum and an IBM DB2 database to Spotfire with similar capabilities. In addition, there are also Microsoft SQL Server Analysis Services compatible cubes available on Hadoop. Examples are Apache Kylin™, Kyvos, and AtScale. The indexing engine, Jethro, also provides a very fast SQL query interface for Hadoop.

FUNCTION SUPPORT	
SQL DATABASE	HADOOP
Wide, often including functions like binning.	The number of supported functions are in general not on par with most SQL databases but they are growing over time. Data connectors hold a mapping of Spotfire functions to functions in the connector's data source. If you believe a certain function should be mapped but isn't, please share your idea with us on the Spotfire idea portal, https://spotfireideas.tibco.com/portal_session/new.

For more details on function support, please see Spotfire online documentation and browse to Connectors. https://docs.tibco.com/products/tibco-spotfire-analyst-7-8-0

WE USE THE SPOTFIRE UNPIVOT DATA TRANSFORMATION A LOT. IS IT POSSIBLE TO UNPIVOT BEFORE END USERS LOAD DATA BY PUSHING THE UNPIVOT OPERATION DOWN TO THE DATABASE BY CONFIGURING IT AS PART OF AN INFORMATION LINK OR A DATA CONNECTION?

The Spotfire unpivot data transformation is available when adding a data source, for example an information link, to your analysis. It's also possible to add later on as a separate step. Pivot and unpivot operations can be designed in information links using the Condition/Pivot dialog or using custom SQL. Use a custom query in data connections.

WHAT OPTIONS ARE AVAILABLE WHEN DOING A UNION OR INSERTING ROWS IN SPOTFIRE?

The Spotfire insert rows capability has long been a convenience for users wishing to append data to an existing data table, such as adding the current month of sales transactions. With Spotfire 7.8, the recommendation engine makes the process even more intuitive for users to add data as rows when applicable and match columns automatically. Additional columns can be added to the final data table, and Spotfire keeps track of new and existing origin columns. If it would be valuable to add rows using union intersect or union minus, please let us know in the Spotfire idea portal. Custom queries or SQL can also be used. Also, see the section about unpivot above.

RELATIONS IN SPOTFIRE SEEM TO BE BASED ON INNER JOINS. IS THERE A WAY TO CHANGE JOIN TYPE FOR RELATIONS?

Relations are used for brush linking (marking across visualizations) and for filtering across data represented as separate data tables using, for example, the filter panel. Data tables can be defined (in-memory or in-database) using connectors. Default inner joins are helpful for combining two fact tables in a single visualization. If inner-join behavior is not ideal, users can define joins in the information link or data connection and return a single resulting table to Spotfire. By selecting tables and using custom or existing relations in the data source, Spotfire views all tables as one virtually joined data table. This eliminates the need for Spotfire relations because table relations have already been defined in the data source/data connection. End users can analyze one virtually joined table dimension free. Other options include the Insert Columns feature where columns can be appended to the main table from the secondary table. Join types can be specified here.

HOW CAN I INCLUDE OR EXCLUDE DATA ROWS FROM A DATA SET BEFORE LOADING IT INTO SPOTFIRE? FOR EXAMPLE, BASED ON VALUES IN A TRUE OR FALSE COLUMN?

When connecting to a data table, loading time and memory space can be saved by reducing the volume of data based on filters. In information links, use the Filters dialogue to set up where statements in the resulting SQL query. In data connections, use a custom query. For either type of connection, you can specify prompts that will alert the user to set filter(s) on predefined columns.

PROMPTS

Information links and data connectors both support prompts that can be used to create selection dialogs. Dialogs make it easy for business users to choose unique values or ranges, such as a date range, that are implemented as where filters.

PARAMETERIZED CUSTOM QUERIES

Data connectors support parameterized custom queries that programmatically pass values based on interactivity in your analysis or SQL query. This can be used to limit data based on, for example, document properties or Spotfire domain, group, and user variables. The latter is useful for filtering out personal data from a generic data set.

DATA ON DEMAND

Data-on-demand is a popular way to limit data based on user interactions with marking, filtering, and actions, both before initial loading and during an analytic session.

WHAT IS THE RECOMMENDATION FOR IMPLEMENTING ROW LEVEL SECURITY IN COMBINATION WITH VERY LARGE EMBEDDED DATA?

Because of the time it takes to load an analysis file, scheduled updates are used. Scheduled updates can load the analysis file into the Web Player memory during off hours so that it's ready for business users the next day. However, since data is loaded by a scheduled updates user (an administrator), all data is loaded at one time. In many cases, each user is only allowed to see their personal slice of the complete data set immediately.

To combine preloading of large embedded data sets using scheduled updates with personal data, a lookup table and Spotfire user identities are used in combination with the complete data set. This is referred to as "personalized information links."

The idea is that a join is much quicker to perform than data loading. When a user logs in and opens the analysis file, the first thing that happens is that the complete data table is joined with the user's rows in the lookup table. The result is a much smaller data set, only containing the rows that are left after the join, and only personal data.

Note: This is a solution for the thousands of consumers using the Spotfire Consumer client on the Web Player. Users with authoring licenses using the Spotfire Analyst or Business Author clients can edit the analysis files configuration and gain access to the full data set.

WHAT OPTIONS ARE AVAILABLE FOR WRITE BACK OF INFORMATION INTO A DATA SOURCE FROM SPOTFIRE?

Even though Spotfire is by default a read-only platform, it's very common to configure it to write back information into data sources. A common use case is to tag rows with comments that group rows together. This is very useful, for example, when multiple teams perform different parts of an analysis. The first team can then tag the rows of interest to them, which makes it easy for the next team to know where to start the continued analysis.

For more information about different ways of implementing write back, please see this Spotfire Community section. Techniques include using an IronPython script, stored procedures in information links, and R or TIBCO® Enterprise Runtime for R (TERR) data functions.

WHAT'S THE RECOMMENDED WAY OF CONNECTING TO STAR SCHEMAS?

Even though information links can be used to connect to star schemas, it's much easier to connect using data connectors. The table below summarizes some of the differences.

CAPABILITY	INFORMATION LINKS	DATA CONNECTORS
RELATIONS	Relations are (re)created manually in Information Designer.	Joins that are defined in the database are discovered and recognized by Spotfire. Users configuring data connections can load relations from the data source with a couple of mouse clicks. The relations automatically become part of the (in-database) view to be analyzed. If data is imported into memory, the selected relations will define a join that is performed in the data source. The resulting data table is then loaded into memory. Since the result is only one data table, visual analytics becomes very easy to do with no need for additional add columns operations or relations.
DRIVERS	JDBC drivers only. Drivers are needed on the Spotfire Server only.	ODBC drivers, ADO .Net drivers, or possibly no driver at all is needed, depending on the connector used. Drivers are needed on the Node Manager/Web Player server and on the Windows clients running Spotfire Analyst.

WHAT CAN I DO TO IMPROVE SPOTFIRE DATA LOADING SPEED?

You are probably aware of the general factors impacting your data loading speed. Examples are hardware resources, database/cluster size, query engine performance, query complexity, network bandwidth, number of concurrent users, and use of the latest drivers recommended by database vendors. We recommend making yourself aware of the capabilities and settings in the Spotfire platform. These improve the time it takes for data to be loaded and ready for visual data discovery. Some of these capabilities and settings are highlighted below. They are divided into loading data using live queries and in-database data tables, and loading data as extracts into the Spotfire in-memory data engine.

LIVE QUERIES/IN-DATABASE

The performance of big data analytics using live queries against in-database data tables is directly dependent on how fast the database's SQL or MDX query engine can process Spotfire queries.

If the use case is a dashboard, the user interactivity with the analysis file is often lower with modest load on the data source's query engine. In this case, the Spotfire live query cache is extremely useful because a dashboard tends to reuse the same queries more often with no need to push queries to the data source. Users are simply using visualizations more or less as they are configured by the author of the analysis file.

If data in the data source is not live updated, or updated during the use of the analysis file, or if it's not part of the use case to analyze live data, it's recommended to increase the default time of the live query cache to 60–120 minutes. This time will ensure that even sporadic interaction with the dashboard and the resulting queries will make use of the live query cache.

IN-MEMORY

The Spotfire in-memory data engine ingests data as fast as it's delivered by database drivers, the data source APIs, or the file reader. There are however capabilities and settings to consider.

Data transformations that are part of the data loading step (part of the data source), or applied as a following data transformation step, take time to process. Reviewing data transformations might speed up data loading.

When loading star schemas, you could consider joining the tables you need in the data source or ingesting them into the Spotfire data engine as separate tables and then using Insert Columns to join reference data to the main table. Connectors by default flatten/join star schemas in the data source before loading, which makes it easy to work with connectors, but can be time consuming and can increase the size of the table that is loaded.

Narrow data sets in general load more quickly and are more suitable for analytics. Wide data sets (thousands of columns) in general take more time to load. In addition, they often require a potentially time consuming unpivot data transformation to transform them to a narrow data set. Also, consider an unpivot of the data before loading the data (see above).

HOW DO YOU WORK WITH CLOUDERA?

As a leader in the distribution of Apache Hadoop, Cloudera is a strategic partner of TIBCO. We work closely with Cloudera engineering teams to create the optimal native connector to Impala. In addition, we leverage the unique capabilities of Spotfire when accessing data lakes; Spotfire in-memory, in-database, or on-demand access methods to Cloudera Hadoop, and the use of Cloudera's speed optimized Hadoop/Parquet/Kudu for storage and query.

From a business and use case perspective, we're partnered with Cloudera field experts to deliver vertical value propositions and templates for industries like Semiconductor, Oil and Gas, Utilities, and Retail to name a few. Our intelligential property around data structure, data partitioning, metadata, data load, data storage, and data access is unique in a Cloudera + TIBCO Spotfire implementation.

Our Spotfire architects are also trained on Cloudera and AWS. This gives TIBCO Spotfire a competitive advantage and time to market advantage when architecting your mission-critical enterprise analytic platform.

TIBCO®

Global Headquarters
3307 Hillview Avenue
Palo Alto, CA 94304
+1 650-846-1000 TEL
+1 800-420-8450
+1 650-846-1005 FAX
www.tibco.com

02/23/17