INFORMATION TECHNOLOGY (IT)

MULTIPLATFORM DATA ACQUISITION, COLLECTION AND ANALYTICS (MDACA)

BIG DATA VIRTUALIZATION USER GUIDE

Version 3.0

BACKGROUND

REVISION HISTORY

VERSION	RELEASE DATE	AUTHOR	REVIEWER(S)	APPROVER	CHANGES
1.0	04/30/2021	SpinSys-MF	SpinSys-LAF	SpinSys-CJ	Initial version.

TABLE OF CONTENTS

LIST OF FIGURES

1 PURPOSE AND OVERVIEW

The Multiplatform Data Acquisition, Collection, and Analytics (MDACA) Big Data Virtualization (BDV) is designed to provide a single view of enterprise data and conceal the technical complexities of database types, data locations, and data transformations for business owners.

MDACA BDV solution provides a logical data layer that integrates enterprise-wide data across disparate systems and manages the unified data within a single location for centralized data access. MDACA BDV aims to simplify the overall data governance process by eliminating the need for monitoring multiple point-to-point connections.

1.1 What is MDACA

MDACA (Multiplatform Data Acquisition, Collection and Analytics) is a scalable big data suite of applications for data acquisition, storage, and access. MDACA is designed for fast transfer, organization, and management of large volumes of data.

1.1.1 MDACA Big Data Virtualization

See Figure 1 for an image of MDACA Big Data Virtualization.

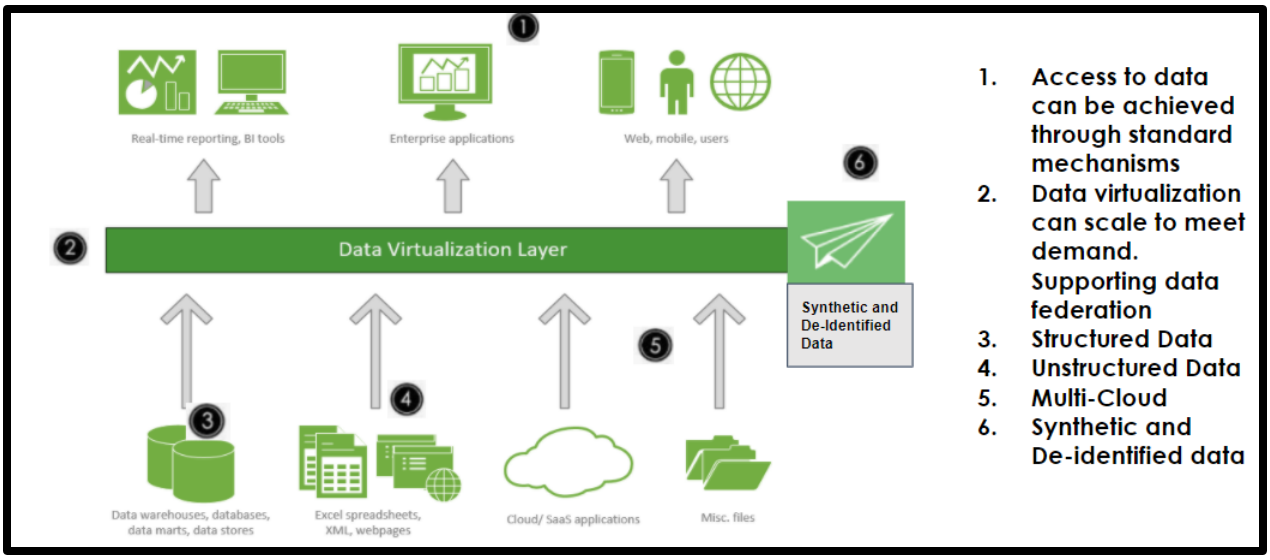

Figure 1: MDACA Big Data Virtualization

1.1.2 Centralized Data Access Leveraging Query Federation

In traditional enterprise Big Data environments, data is stored across multiple databases, systems, and applications. For enterprise level reporting, analytics and application development that rely on the data across disparate systems requires continuous effort, coordination, and data copying and integration activities from various teams in order to achieve business goals and objectives. Additionally, if multiple databases use different technologies (e.g. SQL Server vs. Oracle vs. Postgres), developers and data analysts must learn how to query the databases in database vendor compliant SQL among other logistics and access challenges. See Figure 2.

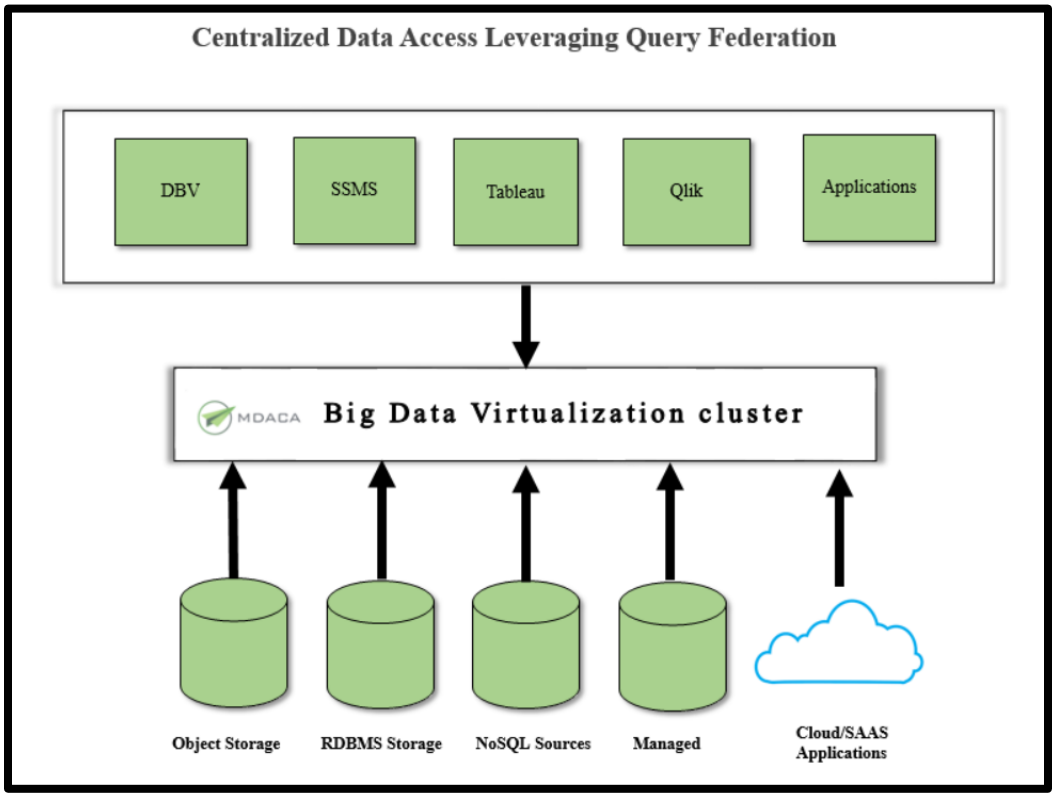

Figure 2: Enterprise Cloud Environments

The MDACA configuration team will coordinate with system owners to configure connections and security access mechanisms. Additionally, the configuration team will collaborate with system owners to define any data security/obfuscations and configure the virtual layer with virtual schemas of the data. These virtual schemas can be applied to one or multiple systems, while continuing to hide the configuration complexities from the end user.

Once the configuration is complete, the process of accessing data across multiple systems becomes standardized, cost effective and efficient. Furthermore, system data standardization and Master Data Management (MDM) are easily implemented and managed.

1.2 Key Features

The ability to efficiently query large amounts of data is imperative for any enterprise organization that manages Big Data. As government and commercial entities continue to evolve and modernize, they must address new challenges as a result. As it becomes more cost efficient to collect and store vast amounts of data across multiple locations and databases, it is equally important to ensure that the tools used to query this data are fast, user friendly, and flexible. MDACA BDV is designed to address this challenge by breaking down the technical complexities through a simplified user interface while simultaneously managing data across a variety of sources within a centralized location.

1.2.1 Ease of Use and Flexibility

MDACA BDV allows enterprise users to easily access data across a wide range of data sources and database types through query federation. End users, business intelligence (BI) applications and standard applications are able to access data from multiple systems within a single query. For example, the MDACA BDV is able to combine data that is stored within an S3 storage object with data that is stored in multiple SQL relational databases (i.e. Oracle, SQL Server, and PostgreSQL) into a single query.

1.2.2 Cost Effective

MDACA BDV is designed to reduce costs by masking the complexities of the data and system architecture through a user friendly interface. The typical approach to querying data stored across multiple technologies (e.g. Microsoft Excel, SQL databases and/or Big Data technologies such as Kafka and Elastic Map Reduce (EMR)) is to develop applications that extract and transform data into a single location and then requiring users to query the combined data within the designated location. This method requires a high level of effort from multiple teams, as ETL development and cross-collaboration across development teams must be continuously upgraded to support systems and tools across the enterprise.

1.2.3 Security and Auditing

MDACA BDV enables access and policy management for your data by fully integrating with enterprise authentication and authorization through support of both Kerberos and standards based Single Sign-on (SSO), including Security Assertion Markup Language (SAML), OpenID Connect (ODIC), etc. Furthermore, MDACA BDV offers advanced and configurable audit and logging capabilities.

1.2.4 Enterprise Data Privacy

MDACA BDV allows controlled access to data at the source. With advanced row-level filtering and dynamic column-masking policies, filters and data masks can be set for specific users, groups, and conditions. With column-level masking, sensitive information is contained to the environment while data remains at the source. Controlling and managing access to data at the source eliminates redundant data copies for specialized data marts where users, specifically within the research analytics space, would otherwise be restricted from accessing sensitive data protected by privacy policies such as Health Insurance Portability Accountability Act (HIPAA) and Sarbanes-Oxley.

1.2.5 Extendable and Customizable Features

MDACA BDV offers a highly parallel and distributed query engine, built from the ground up, to provide efficient and extendable low latency analytics. Furthermore, MDACA BDV provides an American National Standards Institute (ANSI) SQL compliant query engine that is capable of supporting a wide range of query needs, including:

- Interactive speeds
- Massive multi-hour batch queries
- High volume applications that perform sub-second queries

2 BDV MENU AND PANELS

DbVisualizer is the ultimate database tool for developers, analysts and DBAs. It runs on all major OSes and connects to all major databases.

2.1 Licensing

DbVisualizer Pro is licensed per individual user. A licensed user may run DbVisualizer on different computers and operating systems, regardless of location and even concurrently on more than one computer. For example, you can run DbVisualizer on your computer at the office and on your laptop at home.

DbVisualizer Pro offers enterprise licensing for simplified license management over multiple users, meaning that only one single license key is generated. The software is still licensed per user so the amount of users need to be monitored to not exceed the licensed amount.

2.2 Installing DbVisualizer

- DbVisualizer is a generic JDBC SQL client and a tool of choice for BDV users. It runs on all major Operating Systems (OS) and connects to all major databases. Note that BDV can be accessed using other generic JDBC/ODBC SQL Clients such as AquaStudio or DBeaver.
- You can download DbVisualizer free edition at https://www.dbvis.com/download
- DbVisualizer Pro extends the free edition with a collection of productivity features and is available with premium support or basic support. It expands on a variety of capabilities to include table management, table import/export, query builder, and database scheduling, events, and jobs among others.

2.3 Install Driver

A user must coordinate with a BDV admin to obtain the BDV database driver. See Figure 3.

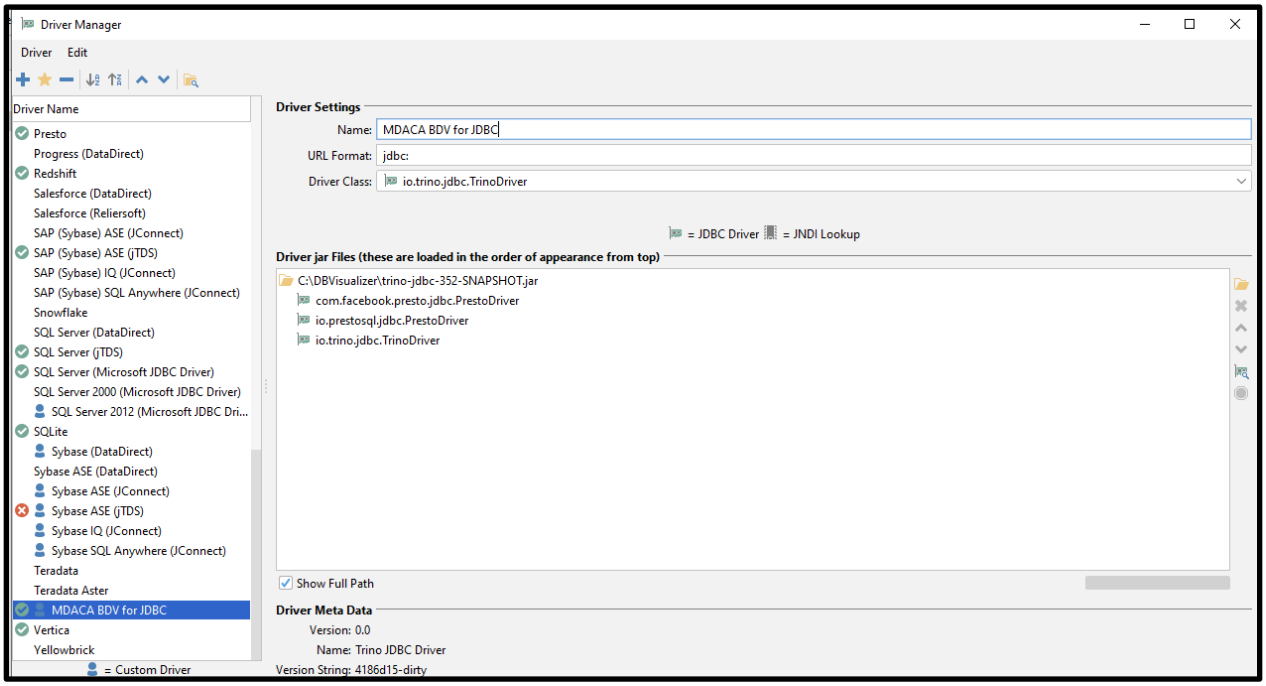

Figure 3: BDV Driver Installation

2.4 Set up the Connection

Please follow the steps below for connecting to MDACA BDV from DbVisualizer. See Figure 4.

- Launch DB Visualizer.
- From DB Visualizer, click “File” and then “Import Settings” and browse for the “mdaca-bdv.jar” file on your local system to install the MDACA BDV connection.

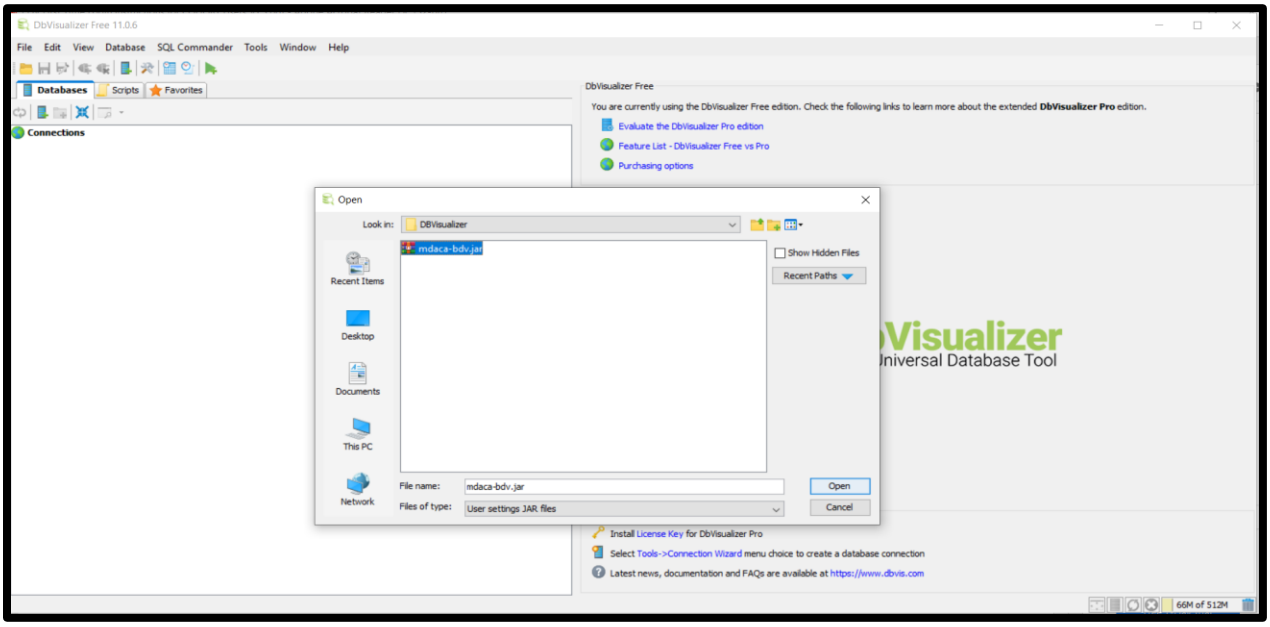

Figure 4: Connection Setup (a)

After DbVisualizer imports the MDACA BDV connection file, it will prompt you to restart the application to take effect. See Figure 5.

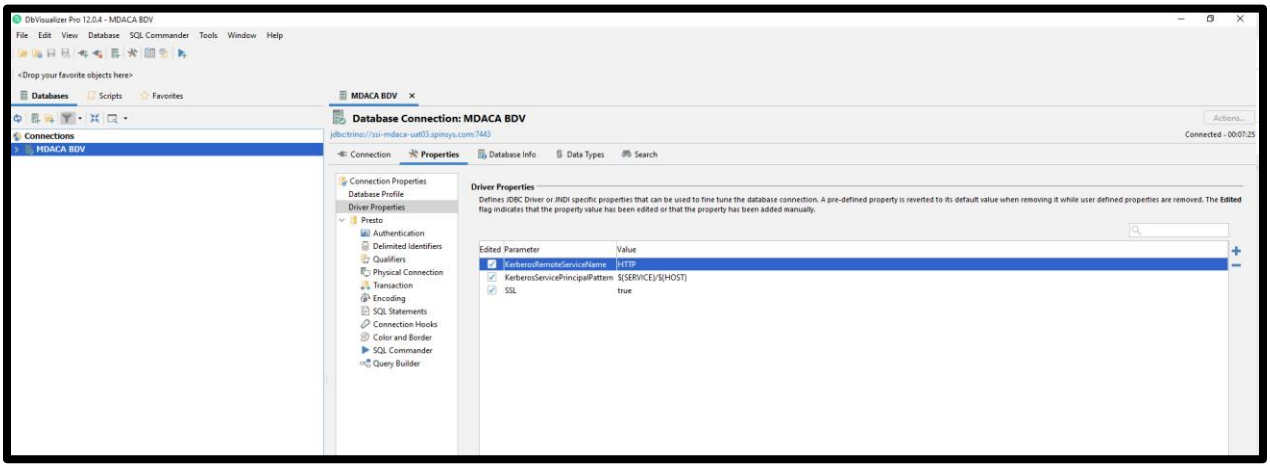

Figure 5: Connection Setup (b)

Once the application has restarted, verify the Driver Properties reflect what is shown in the screenshot above. See Figure 6.

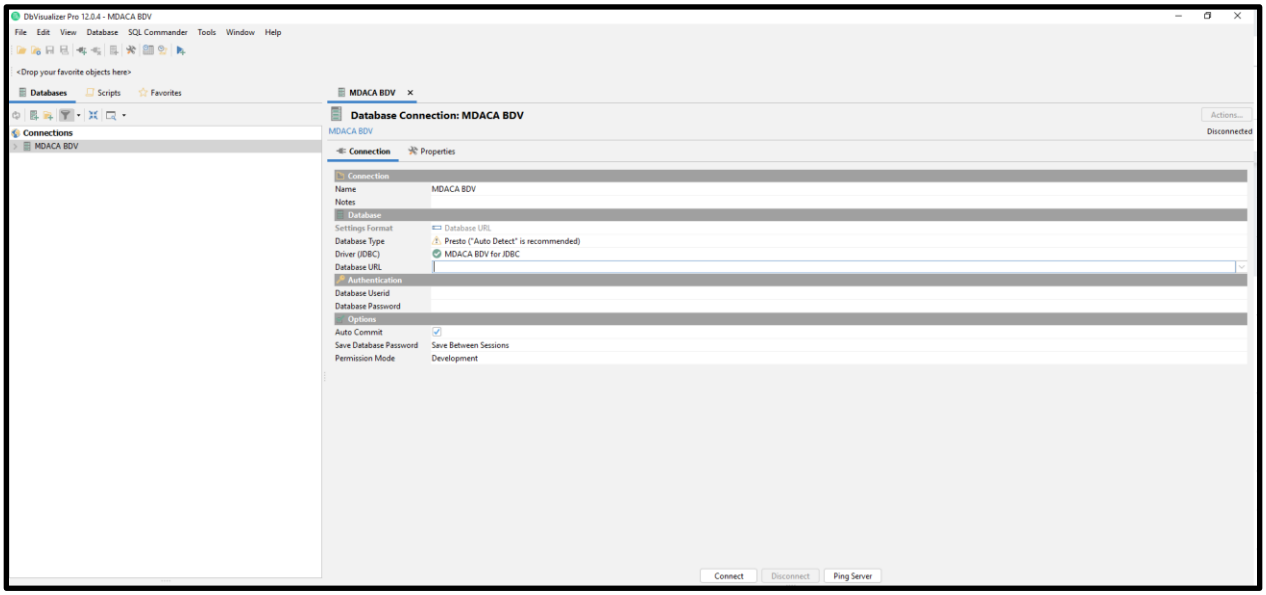

Figure 6: Verify Driver Properties

The “Database URL” and “Database UserId” fields in the screenshot above are required and should be provided to you by your BDV administrator. The “Database Password” field is not used.

Click the “Connect” button to access the BDV catalogs granted to you by your BDV administrator. Navigate back to “MDACA BDV” on the left panel and click on the “+” icon to expand the list of all accessible database schemas, tables, and views.

2.5 Test Connection with a Sample Query

BDV uses a dot notation scheme to identify database tables and views. To properly reference a database table in your SQL, the syntax should follow "SELECT * FROM catalog.schema.table". However in the screenshot above, DbVisualizer provides dropdowns to allow you to filter your queries to use a specific schema within a specific catalog. See Figure 7.

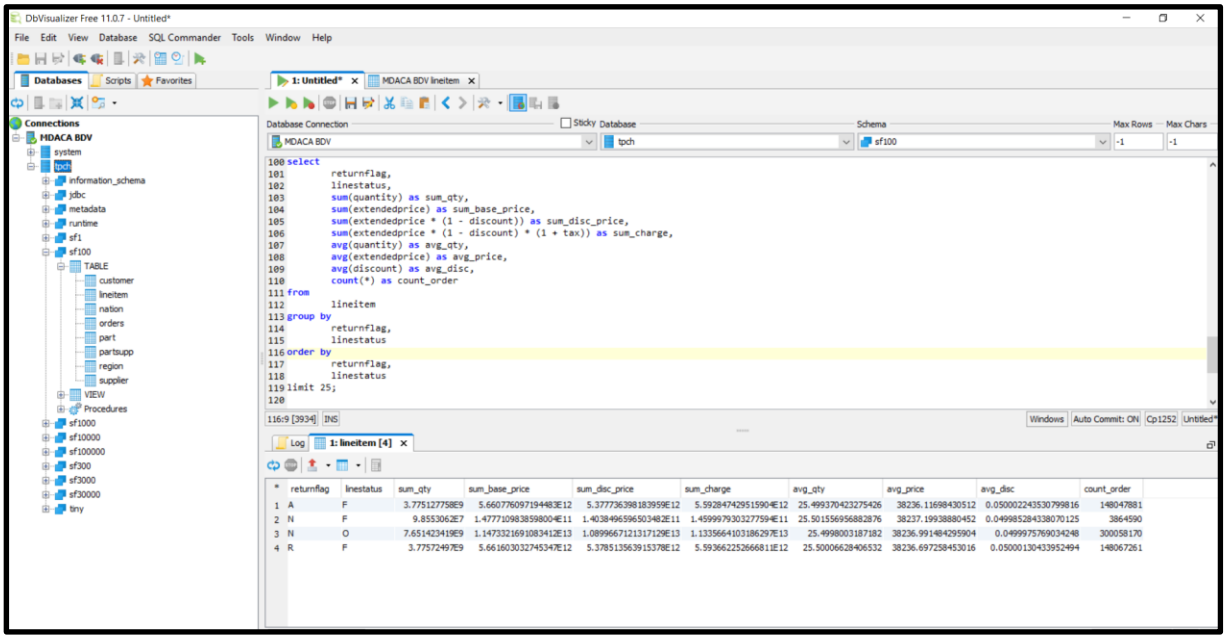

Figure 7: Sample Query

2.6 Cluster Overview

Cluster Overview is a utility for system and database administrators for monitoring BDV queries, query performance and query execution plans. Access to Cluster Overview is granted by the BDV administrator.

The main page has a list of queries along with information like unique query ID, query text, query state, percentage completed, username and source from which this query originated. The currently running queries are at the top of the page, followed by the most recently completed or failed queries. The possible query states are as follows:

1. QUEUED – Query has been accepted and is awaiting execution.
2. PLANNING – Query is being planned.
3. STARTING – Query execution is being started.
4. RUNNING – Query has at least one running task.
5. BLOCKED – Query is blocked and is waiting for resources (buffer space, memory, splits, etc.).

6. FINISHING – Query is finishing (e.g. commit for autocommit queries).
7. FINISHED – Query has finished executing and all output has been consumed.
8. FAILED – Query execution failed.

The Cluster Overview page displays every query submitted to BDV. Performance analytics for each query can be reviewed by clicking on its ID (highlighted above). See Figure 8.

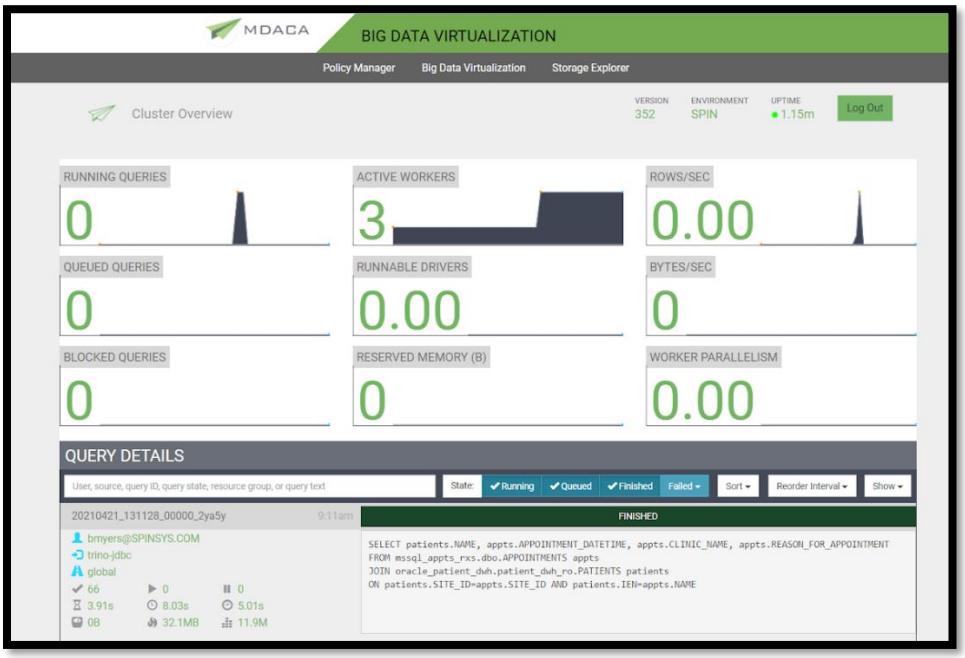

Figure 8: Cluster Overview

The performance analytics captured for each query generally detail how long the query took to execute and the amount of system resources consumed during its execution. The light blue tabs provide further insight into the query execution plan and stages.

The administrator may choose to kill a query that is either taking too long to execute or consuming excessive resources on the system. Certain queries may be terminated based on user request. See Figure 9.

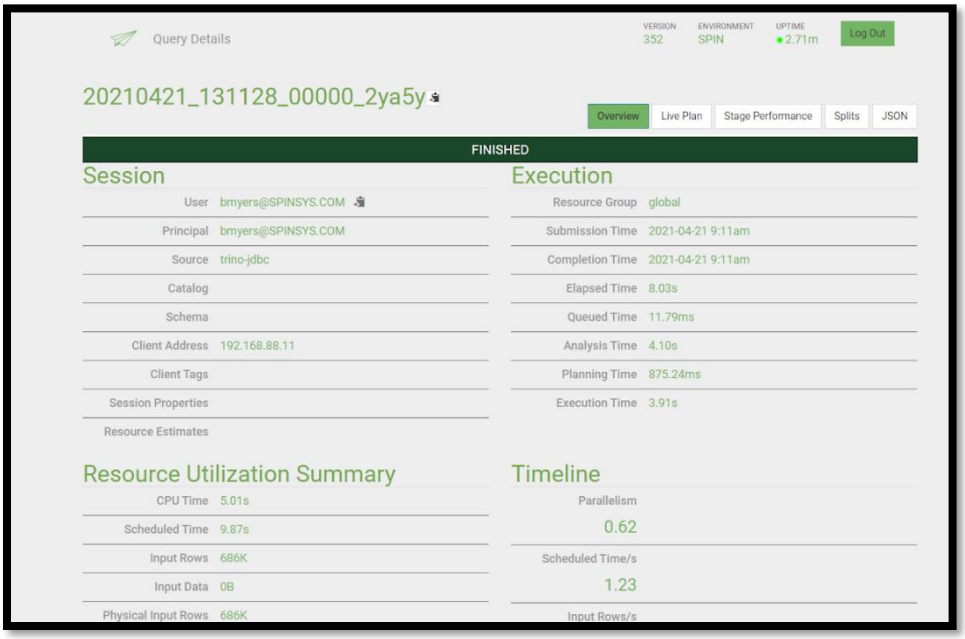

Figure 9: Cluster Overview

2.7 Refactoring EDV Queries for BDV

Although both BDV and its predecessor Enterprise Data Virtualization (EDV) support ANSI SQL, there are subtle differences in syntax when using functions and expressions with BDV since the underlying products are different. The following section should aid the analyst in mapping legacy EDV functions to new BDV functions. BDV adds functions and expressions that were previously not available within EDV. The spreadsheet included in Appendix A of this user guide should ease the transition to BDV. Additional query refactoring may be needed to object references (schemas and table names) since the hierarchical object tree under DBVisualizer appears different when connected to BDV.

2.8 Reporting Issues and Technical Support

In the event you encounter a technical issue when connecting to the Virtual Database or when upgrading DbVisualizer, please submit a ticket to our service desk at edv_support@spinsys.com. All BDV account setup issues should also be sent to the same support email address listed above.

APPENDIX A ATTACHMENTS

1	EDV_BDV_Function_Mapping	 edv-bdv-function-mapping.xlsx