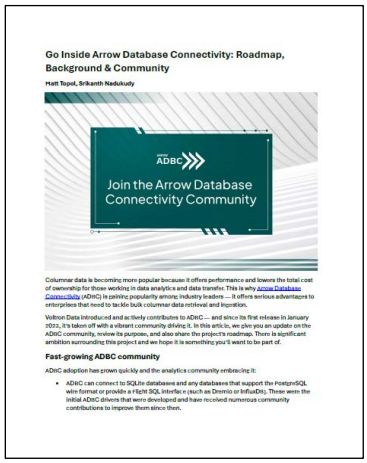
VOLTRON DATA

carahsoft.



Go Inside Arrow Database Connectivity: Roadmap, Background & Community

Matt Topol, Srikanth Nadukudy

ADBC

Join the Arrow Database Connectivity Community

Columnar data is becoming more popular because it offers performance and lowers the total cost of ownership for those working in data analytics and data transfer. This is why Arrow Database Connectivity (ADBC) is gaining popularity among industry leaders — it offers serious advantages to enterprises that need to tackle bulk columnar data retrieval and ingestion.

Voltron Data introduced and actively contributes to ADBC — and since its first release in January 2023, it's taken off with a vibrant community driving it. In this article, we give you an update on the ADBC community, review its purpose, and also share the project's roadmap. There is significant ambition surrounding this project and we hope it is something you'll want to be part of.

Fast-growing ADBC community

ADBC adoption has grown quickly and the analytics community embracing it:

- ADBC can connect to SQLite databases and any databases that support the PostgreSQL wire format or provide a Flight SQL interface (such as Dremio or InfluxDb). These were the initial ADBC drivers that were developed and have received numerous community contributions to improve them since then.

D2 LyãλŘŜ ! NJNJ2ø 5l·űl·ōl·äŜ /2yýŜŏűλØλűèY w2l·ŘYˈl·LJΣ . l·Ŏ1∃NJ2dzyŘ ƨ /2YˈYdzyλűè

Thank you for downloading this Voltron Data resource. Carahsoft is the distributor for Voltron Data's AI and ML solutions available via NASA SEWP V, ITES-SW2, NASPO ValuePoint, and many more contract vehicles.

To learn how to take the next step toward acquiring Voltron Data's solutions, please check out the following resources and information:

For Voltron Data overview:
carah.io/Voltron-Data

For upcoming events:
carah.io/Voltron-Events

For additional resources:
carah.io/Voltron-Resources

For additional Artificial Intelligence:
carah.io/ai-solutions

To set up a meeting:
VoltronData@carahsoft.com

To purchase, check out the contract vehicles available for procurement:
carah.io/procurement

For more information, contact Carahsoft or our reseller partners:
VoltronData@carahsoft.com | 888-606-2770

Go Inside Arrow Database Connectivity: Roadmap, Background & Community

Matt Topol, Srikanth Nadukudy

Columnar data is becoming more popular because it offers performance and lowers the total cost of ownership for those working in data analytics and data transfer. This is why Arrow Database Connectivity (ADBC) is gaining popularity among industry leaders — it offers serious advantages to enterprises that need to tackle bulk columnar data retrieval and ingestion.

Voltron Data introduced and actively contributes to ADBC — and since its first release in January 2023, it's taken off with a vibrant community driving it. In this article, we give you an update on the ADBC community, review its purpose, and also share the project's roadmap. There is significant ambition surrounding this project and we hope it is something you'll want to be part of.

Fast-growing ADBC community

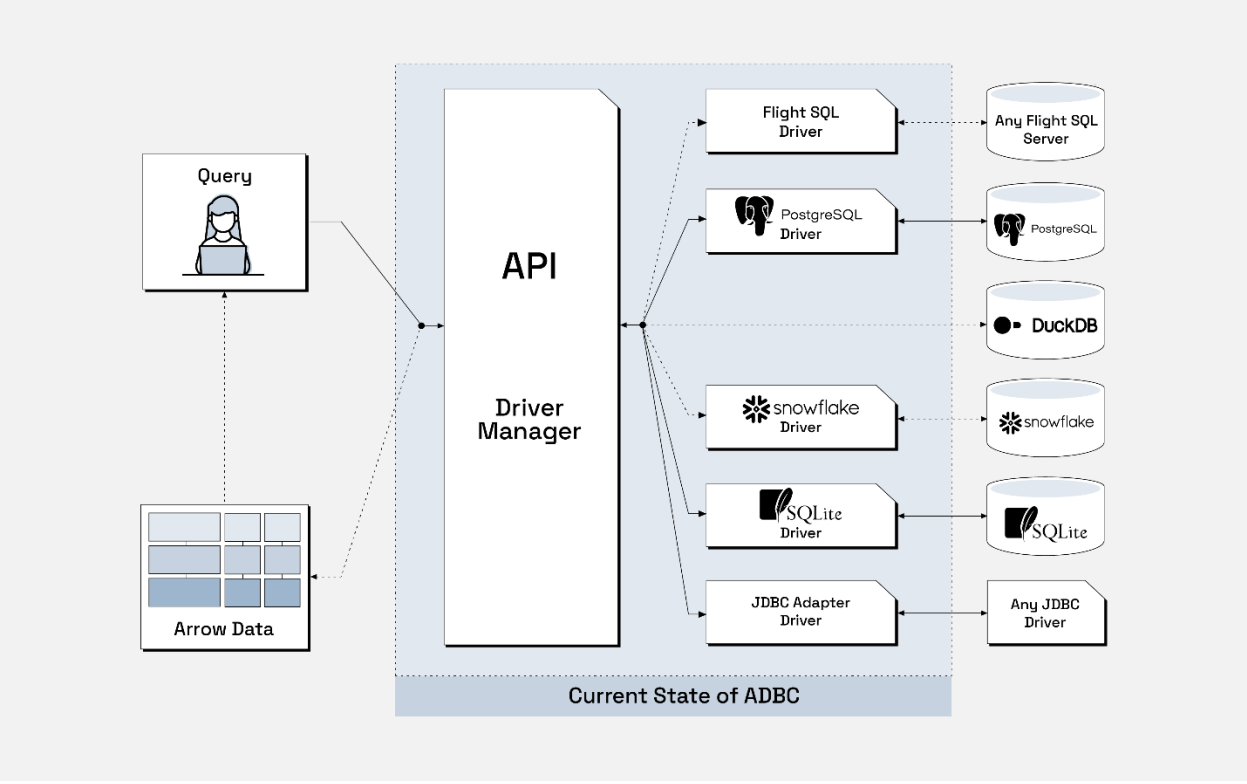
ADBC adoption has grown quickly and the analytics community embracing it:

- ADBC can connect to SQLite databases and any databases that support the PostgreSQL wire format or provide a Flight SQL interface (such as Dremio or InfluxDB). These were the initial ADBC drivers that were developed and have received numerous community contributions to improve them since then.

- Snowflake added support for ADBC (working closely with our team) to make it quicker and easier for their users to connect with preferred languages without needing a bespoke Snowflake connector. Learn more about it in this video from the Snowflake Summit.
- Real time query engine, Deephaven, released an ADBC client with version 0.21 of their product's Python interface, which can be used to extract data from relational database systems that only support ODBC/JDBC access.
- For connecting R applications to databases via ADBC, the adbi package aims to provide a DBI-compliant database access by connecting identically across different databases.
- The team at DuckDB, the fast in-process SQL database known by the same name, has also added support for ADBC. DuckDB is an in-process database, the database itself is the driver.
- Microsoft are the primary contributors of C# bindings and its implementations

What is Arrow Database Connectivity?

ADBC is a standard C API with existing implementations in C, C++, Go, Java, Python, R, C#/.NET, and Ruby, that can provide end-to-end columnar access. Here's a simple diagram that shows the current state of ADBC:



Dashed lines show Arrow formatted data transfers

ADBC provides a standard, Apache Arrow-native client API allowing for individual drivers to be implemented for any data source. An application can submit a SQL query via this API, then the

ADBC driver uses the database-specific protocol to send the query. The database then executes the query and returns the result set in a database-specific way, ideally already in Arrow format. Regardless of the database format, the driver will always return the result set as a stream of Arrow record batches back to the user.

ADBC enables efficient, zero-overhead data access for columnar data and also provides a single, optimized conversion for row-to-columnar data for row-based data sources. Furthermore, the ADBC API provides portability across multiple programming languages.

Why does ADBC matter?

Initially, the Arrow ecosystem used a mix of custom wire protocols to communicate with other databases and lacked a standard database interface. Users were forced to use less performant row-based connectors and either experienced the cost of transposing data between columnar and row-based formats or had to build and maintain their own custom database connectors. There was a need to have some standard way to get and put data from databases using Arrow's native data format. Therefore, ADBC was born.

Like JDBC and ODBC, ADBC helps application developers avoid writing code specific (universal Arrow native client API) to each database and can connect to many different data sources using a unified interface, unlocking data sitting across multiple data systems.

An application can link against the ADBC driver manager and specify, at runtime, the location of the driver for any databases it needs to connect to. If this doesn't suffice, we can a instead statically compile a particular ADBC driver with the API. During execution, the driver manager passes the API calls to the correct driver for the database the application wants to connect to, and the data travels in a columnar format. Application developers can maintain a single SQL interface, saving development and code maintenance times, that connects to existing data sources and also to new data sources that start supporting ADBC in the future.

Roadmap towards ADBC Libraries 1.0 stable version

ADBC aims to provide a vendor-independent API for SQL and Substrait-based database access, providing flexibility to connect to both row-based and columnar data sources.

The ADBC API standard is considered stable, with enhancements being made in a backward-compatible manner based on continued ADBC community adoption and real-world examples. Alongside this for the ADBC Libraries, there are plans to get to better feature parity between all the existing ADBC drivers and also support more drivers. For example, here is a recently merged pull request that implements features in the BigQuery Go client needed to build a new BigQuery Go ADBC driver and another pull request that added a BigQuery C# ADBC driver.

Right now, if you have a system that supports JDBC, there's an ADBC driver that allows you to use ADBC's columnar API instead. Based on that, one upcoming idea on the roadmap is an ADBC driver that wraps ODBC to enable the use of ADBC where it is not yet supported. The API will remain the same for the user, and this provides an opportunity for developers to experiment with ADBC in environments currently restricted to ODBC.

Finally, new improvements to error handling and logging are planned to ensure easy adoption for the community, especially with the complexity added as we develop connectivity for more data sources.

Join the ADBC community!

ADBC is becoming the new standard way to connect to any data source. Promotion and use of open standards make data integration easier. As adoption grows, this enables users to connect to any database, without code rewrites saving developer time and making the data more valuable with such easy access. If you'd like to get involved in the ADBC community and contribute to expanding this standard, please reach out to us via GitHub: http://github.com/apache/arrow-adbc. For more information on ADBC and how to get started, please see the documentation.

Furthermore, if you want to integrate ADBC and open standards like Arrow and Substrait into your data stack, Voltron Data has the experts in using these and other open source software to build composable data systems. If you're interested in learning our approach and working with us, check out our Product page.