Yellowbrick

Migrating from Netezza to Yellowbrick, the data warehouse for distributed clouds

Introduction

Netezza introduced the first data warehouse appliance in 2003. However, Netezza didn't keep up with innovations in hardware and software and didn't make the leap to the cloud. By migrating to Yellowbrick, Netezza customers can achieve 100X performance and 10X scale at a fraction of the cost. This document provides an overview of migration tasks and serves as a guide for further streamlining the process.

Modernizing your data warehouse with Yellowbrick

Yellowbrick Data Warehouse is an advanced, massively parallel (MPP), SQL database designed for the most demanding batch, real-time, ad hoc, and mixed workloads. It can run complex queries at up to petabyte scale across numerous nodes, with guaranteed sub-second response times. Yellowbrick was conceived with the goal of optimizing price/performance. It's not uncommon for customers to see their workloads run tens or hundreds of times faster at a fraction of the cost compared to cloud-only or legacy data warehouses.

Yellowbrick continuously implements new hardware (e.g., NVMe and flash memory) and software (most recently Kubernetes) protocols in an adaptive "cut-through" architecture that ensures best performance in every environment. Yellowbrick combines these advances with smart thinking about storage formats and indexing, and add on top a modern, standards-based database interface that's familiar to users (PostgreSQL) for ecosystem compatibility. The result is a modern, quickly provisioned, and easy-to-use data warehouse that knocks the socks off rivals in price/performance economics and can be deployed across distributed clouds (private data centers, public clouds, and edge networks) – with all instances, databases, and users managed through a simple, unified control plane (Yellowbrick Manager).
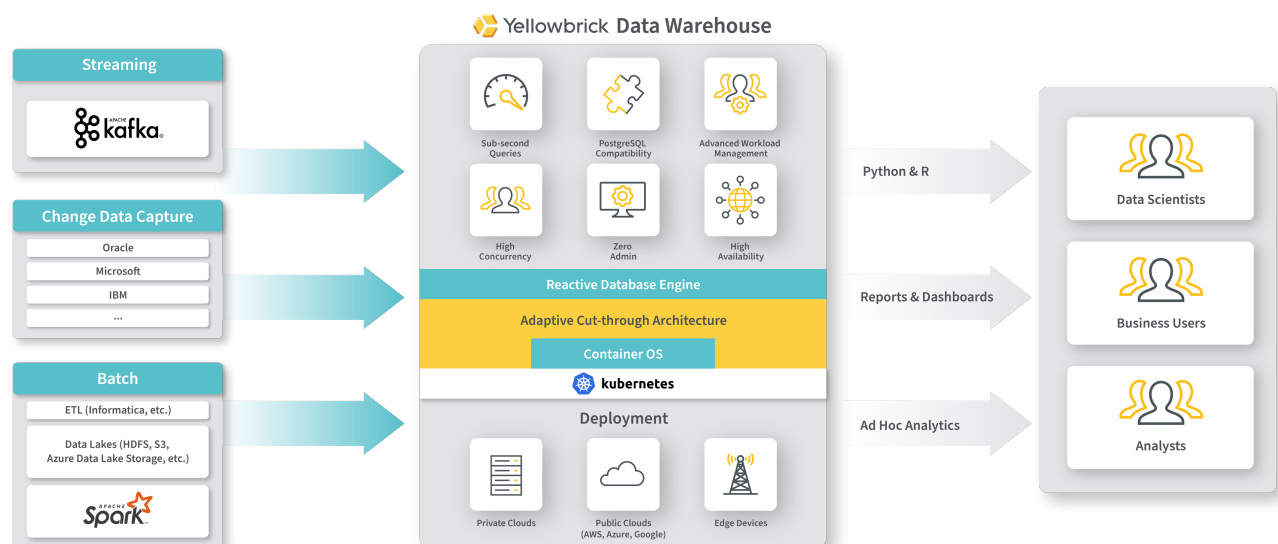
A simple end to end migration

Yellowbrick is designed to make migrating from Netezza fast and simple. The Yellowbrick Data Warehouse:

- Uses the same underlying PostgreSQL dialect as Netezza

- Supports PL/pgSQL stored procedures

- Has a library of functions added for Netezza compatibility

- Has intuitive syntax like Netezza (for example, nzload is ybload and nzsql is ybsql)

- Offers a similar set of ecosystem integrations via standard ODBC, JDBC, and ADO drivers

- Is certified and supported by many of the same enterprise tools you use today, such as Tableau,, MicroStrategy and Informatica

A Yellowbrick solutions architect will help you scope and plan your migration as part of a proof of concept (POC). Once your POC is complete, the solutions architect will help you migrate your schema, data, application logic, ecosystem tools, and users to the Yellowbrick platform.

You'll have a production-ready instance installed and running in under a day and will complete your migration within days or weeks. Your post-migration activities are similarly easy: your team will have very little tuning and configuration, will not have to pay for training and certification, and can get back to day-to-day business quickly.

Yellowbrick is standards-based

The Yellowbrick Data Warehouse supports organizing your data into databases and schemas, just like Netezza and every other PostgreSQL-inspired database. This means you do not have to worry about the migration breaking your applications. You can continue to maintain your database organization as you always have, with one database per tenant. If you want to expand to multiple schemas in a single database, you can do so easily.

The Yellowbrick Data Warehouse includes a query history database that is installed and fully documented. Your queries will be logged and optimized on day one, and you no longer have to endure dumps of in-memory query history or manually configure the NZ_QUERY_HISTORY table as you might have been doing with your Netezza system. Yellowbrick provides a richer, more granular level of detail about all activity within the database and is easy to configure. You can set the retention period for your history with just one click.

Determining the current Netezza DDL

The nz_ddl_database command provides all the information about the DDL in your Netezza system, including table, view, synonym, and sequence definitions; user and group role creations; stored procedures; user-defined functions; and much more.

To extract the DDL for each database, execute the following command:

```
$ nz_ddl_database <database> > <database>.ddl
```

One of our solutions architects will perform an inventory of your DDLs and map them appropriately to Yellowbrick, according to best practices and using automated conversion tools.

Migrating table definitions

Migrating your table definitions is a seamless process. Many Yellowbrick customers move the Netezza DDL (CREATE TABLE) with absolutely no change to their data types. In addition, you will have the opportunity to improve your table distribution with the following options:

- **Replication** for all tables that compress to a gigabyte or less
- **Hash-based co-location** of all tables greater than a gigabyte on a distribution column that links all the tables
- **Randomly distributed** tables for other use cases

Yellowbrick also includes the following table features that enable you to further optimize your database:

- Table clustering and sorting that is like Netezza zone maps, but more granular
- Table partitioning, which is unavailable in Netezza; this feature reduces resource consumption for very large aggregates and supports fact- to-fact joins that are impossible to execute on a Netezza system

Migrating view definitions

Because Yellowbrick is based on PostgreSQL, all your valid SQL operations and database views will run as is.

Migrating stored procedures

Yellowbrick supports PL/pgSQL stored procedures. A Yellowbrick solutions architect will perform an impact analysis of your query history to determine which, if any, stored procedures you should migrate (that is, which stored procedures your queries are using).

Exporting the data

You export Latin as well as Unicode data from your Netezza system with one of these options:

- Direct streaming from Netezza into Yellowbrick (requires giving Yellowbrick necessary permissions)

- Exporting Netezza data to CSV files. You can do this on a Windows or Linux client using the following methods:

    - nz_backup

    - nzsql

    - SQL for remote external tables

Data will export in ASCII-delimited format that can be compressed.

Importing the data

If you exported the data to a CSV file (as opposed to direct streaming), you will import it using the **ybload** tool. **ybload** is intuitive like Netezza **nzload**, but with dramatic improvements. It is designed for parallelization, so you will not need to chunk files manually, and it maximizes the 10Gb network connection shipped with Yellowbrick. **ybload** also compresses data before sending it to Yellowbrick and can load multiple files at once. For example, users can run one **ybload** with a thousand files as input instead of invoking **nzload** a thousand separate times. These optimizations make loading simple and possible to dramatically accelerate load speeds compared to Netezza.

Yellowbrick can also load data directly from cloud-based object storage that's S3-compatible. The platform supports compressed or un-compressed files. If you are an HDFS user storing data in Hadoop, Yellowbrick uses a Spark-based importer that can consume most common Hadoop file formats, such as Apache Parquet or Avro.

Migrating user-defined functions (UDFs)

Netezza systems often have an Oracle compatibility toolkit installed that provides a set of UDFs. Typically, customers use only a small number of UDFs, and the Yellowbrick Data Warehouse includes many of these functions out of the box. A Yellowbrick solutions architect will perform an impact analysis to see if there are any functions that the Yellowbrick Data Warehouse does not provide that are being used by queries.

Users, groups, and security configuration

Like Netezza, Yellowbrick supports local or LDAP authentication methods. Creating database users (or roles) is the primary mechanism for managing local database security. After creating users, you can set up access control by granting or revoking privileges on databases, schemas, tables, columns, and stored procedures.

Users, groups, and security configuration

Like Netezza, Yellowbrick supports local or LDAP authentication methods.

Creating database users (or roles) is the primary mechanism for managing local database security. After creating users, you can set up access control by granting or revoking privileges on databases, schemas, tables, columns, and stored procedures.

Use the following SQL commands: CREATE ROLE, GRANT, and REVOKE.

Yellowbrick LDAP configurations support directory service providers such as Active Directory (AD) for OpenLDAP. LDAP, LDAPS (LDAP over SSL), and LDAP+TLS configurations. This is an improvement over Netezza LDAP integration, which is limited to just authentication and requires Netezza customers to manually create USERS or GROUPS and can result in mismatches to the LDAP service.

Yellowbrick provides an alternative to creating non-superusers manually in the database by synchronizing the users and groups stored in an LDAP directory and creates users and roles with the same credentials. You can synchronize specific users and groups by defining search criteria and filters.

Redirecting users and applications

You can connect ODBC, JDBC, and ADO.NET applications to a Yellowbrick database using standard PostgreSQL drivers you download and install. This means you do not have to purchase new licenses to redirect your existing applications. More importantly, third-party tools (such as BI, ETL, and analytical tools) that are certified for PostgreSQL drivers will automatically be certified for the Yellowbrick Data Warehouse.

Yellowbrick ships the open-standard PostgreSQL versions of these Windows or Linux drivers with every instance. ETL tools leverage **ybload** and **ybunload** for high-speed data movement in and out of Yellowbrick as described in the "Importing the data" section.

Simplify operations with intelligent automation

Yellowbrick is designed to deliver high performance to mixed workloads that modern enterprises require. It can support thousands of users running mixtures of real-time transactional atomic inserts; bulk loads of data; ELT workloads; interactive queries; ad hoc queries; and long-running, batch-oriented reports. While today's mixed workloads require a high-performance solution, as the number and diversity of workloads increases, it is even more critical that you have the ability to specify, control, and automate the work that matters most. Yellowbrick gives you this control with:

- Real-time visibility into workload performance, with alerts, that ensures you know immediately when workloads are struggling. Historical workload reporting also helps you manage workloads effectively across the enterprise.

- Workload management reporting that enables you to see if the performance of different workload groups is meeting business requirements.

- The ability to reprioritize any workload on the fly so you can adapt to unexpected changes in the business environment.

Automating workload management in Yellowbrick is simple and can be done via the Yellowbrick web interface or SQL ALTER commands.

Automated monitoring

Yellowbrick includes 24x7 predictive remote monitoring technology that enables support engineers to handle day-to-day monitoring activities. Yellowbrick support engineers review real-time diagnostic data, provide recommendations, proactively address potential issues, and deliver prompt incident resolution.

You can monitor the platform yourself, instead of or in addition to using the Yellowbrick service. You can do this through automated alerts that easily integrate to the endpoint tool of your choice. This is a simplified experience instead of manually configuring each nzevent you want to monitor.

Increase visibility

Platform views give you comprehensive, real-time visibility into Yellowbrick. Monitoring is performed continuously without affecting platform performance and is thoroughly documented so you can find what you need, when you need it.

Summary

Migrating from Netezza to Yellowbrick, the data warehouse for distributed clouds, is fast and painless. In days or weeks you can have a modern data warehouse that delivers 100X performance at a fraction of the cost.

To learn more about Yellowbrick, visit **yellowbrick.com** to book a demo or sign up for a free test drive.