

Digital Transformation with Data Fabric – Watson Query/ Data Virtualization

- 1. Overviews*
- 2. Technical Details*
- 3. Typical Use Case Patterns*

Piotr Mierzejewski
Director Db2 & BigData
IBM Data and AI

Goal: Be able to expose any or all of organization's critical data assets no matter the location, platform, technology, or schema.

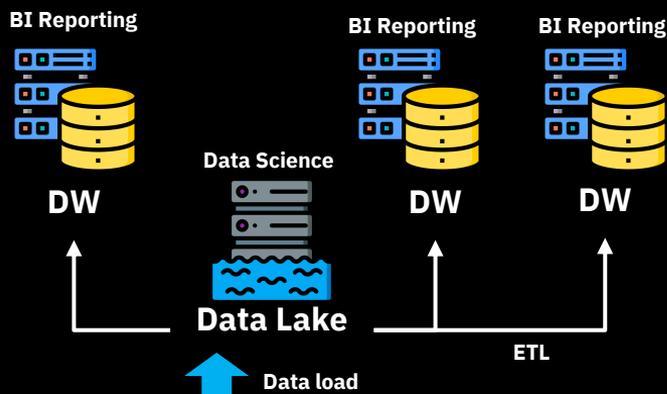
Query 2 or 2,000 data systems with a single query as One



The Vision of IBM Watson Query

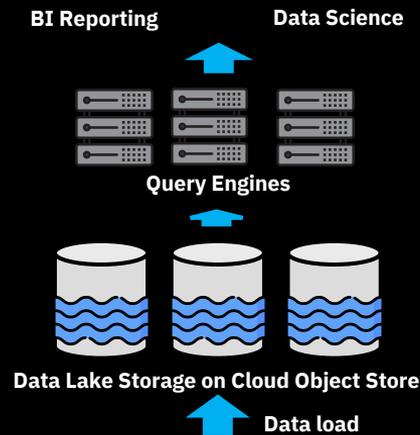
- Ultimately, the goal of Watson Query is to provide the data management experience in Cloud Pak for Data
- Specifically, with a focus on Big Data and how it has evolved in recent years

Classic Big Data Architecture



- Requires many complex data pipelines
- Difficult to scale and maintain
- Duplicates of data
- Isolated BI and Data Science environments

Modern Big Data Architecture



- Scale compute and storage separately
- Handles unstructured and structured data
- Minimal data movement and pipeline
- Single environment for BI and Data Science

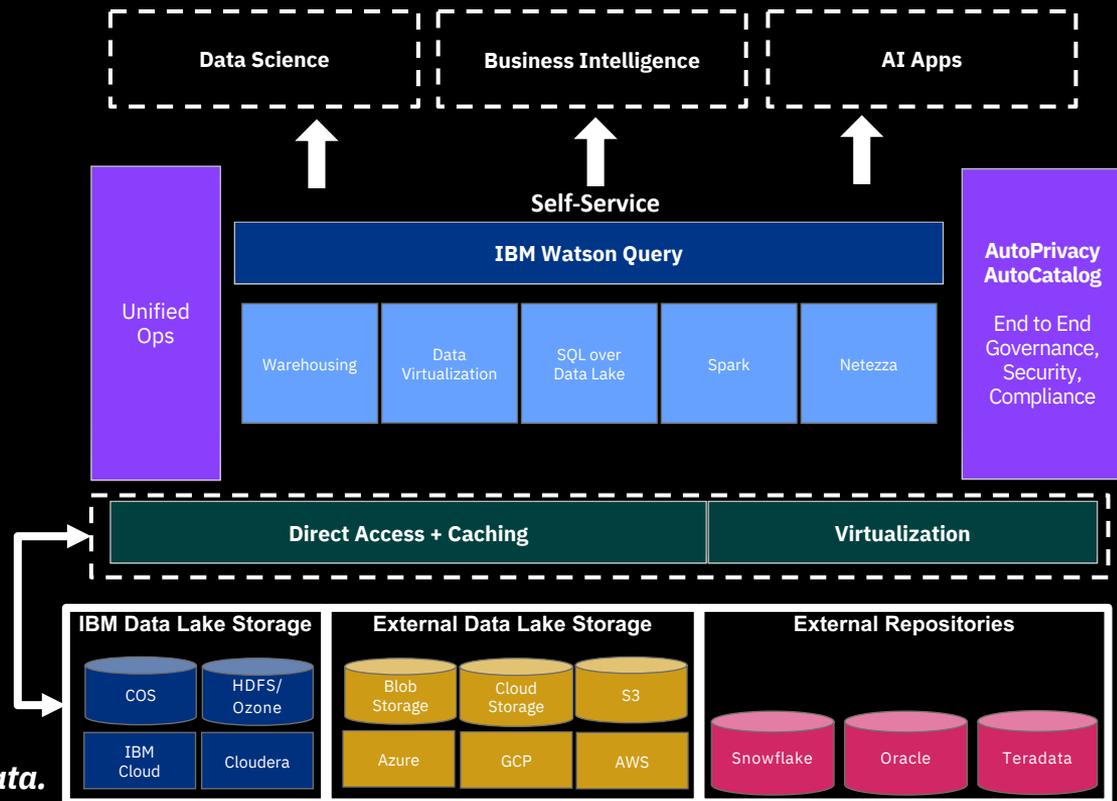


IBM Watson Query

A universal query layer that automates how you access, update and unify data across any source or type (clouds, warehouses, lakes, etc.) without the need for data movement or replication. With intelligent performance optimizations, peta-byte scale and visual query building experiences, it abstracts the complexity of multiple query engines to provide simplified self-service data across an organization

Distributed Query Experience

- **One Query Experience.** Multiple capabilities from virtualization to warehousing
- **End to End Integrated Governance** with AutoCatalog and AutoPrivacy
- **Open Data Formats** to work with data on any cloud and on-prem
- **Petabyte scale** landing, exploration, shaping, governance, and AI
- **Cloud-native** with **consumption pricing**, **instant-on**, and **available** across any zone
- **Fully elastic** with separate compute and storage
- Unified access **without data movement**
- **Hybrid cloud:** Available to deploy on multiple clouds (on prem, private, public)

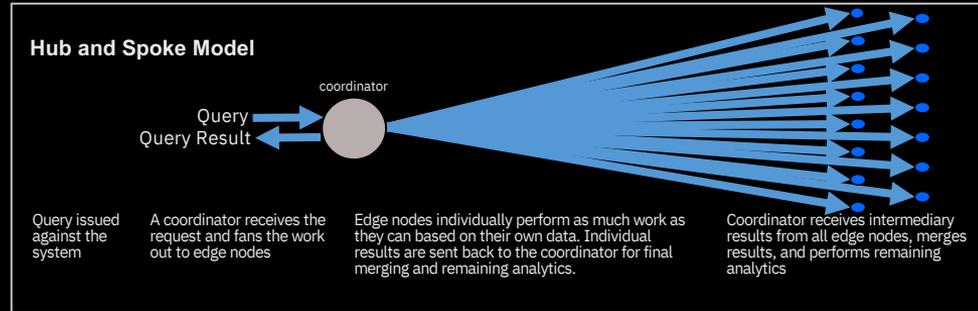


**Any data.
Anywhere.**

Key Architectural Differentiation

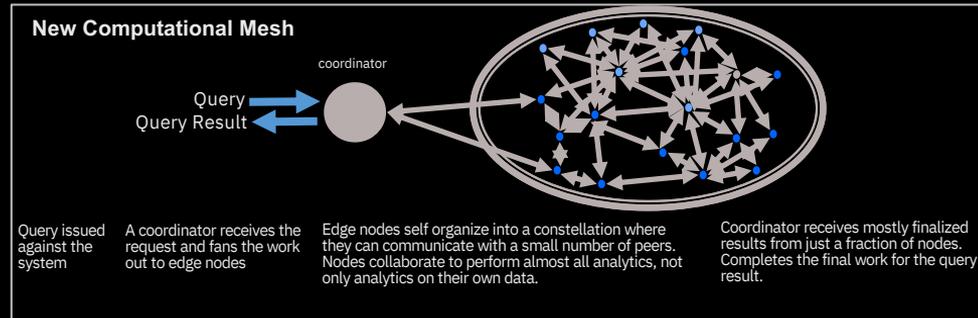
Hub and spoke execution models:

- Lacks scalability
- Performance constrained
- Basis for Federation and our competitors



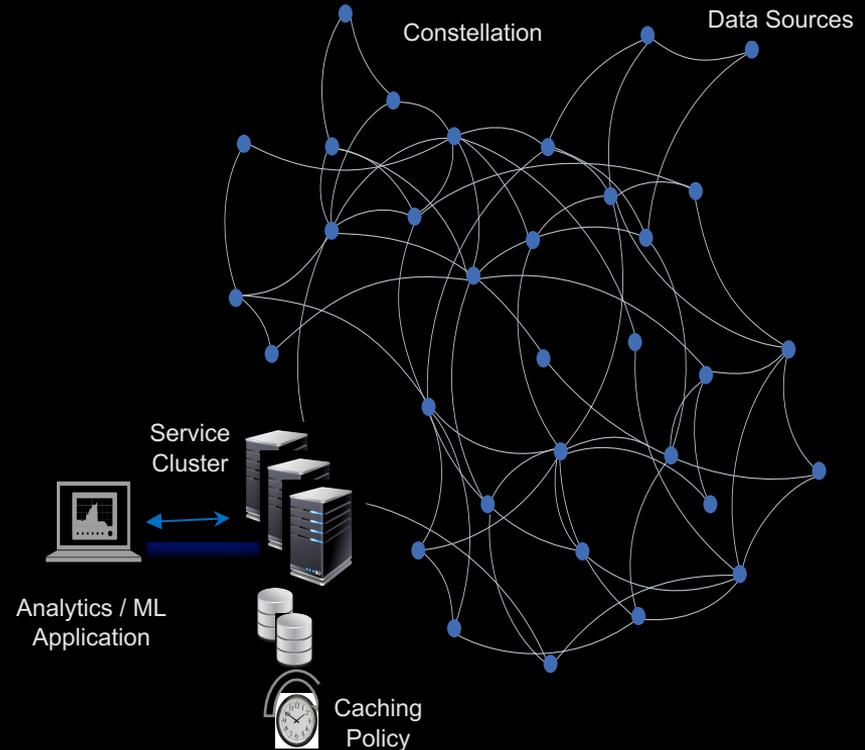
IBM is first to market with a parallel processing model:

- Theoretically unlimited scalability
- Ease of addition/removal of sources
- Execution pushed down into the constellation mesh



Mesh: Remote Connectors and Data Discovery

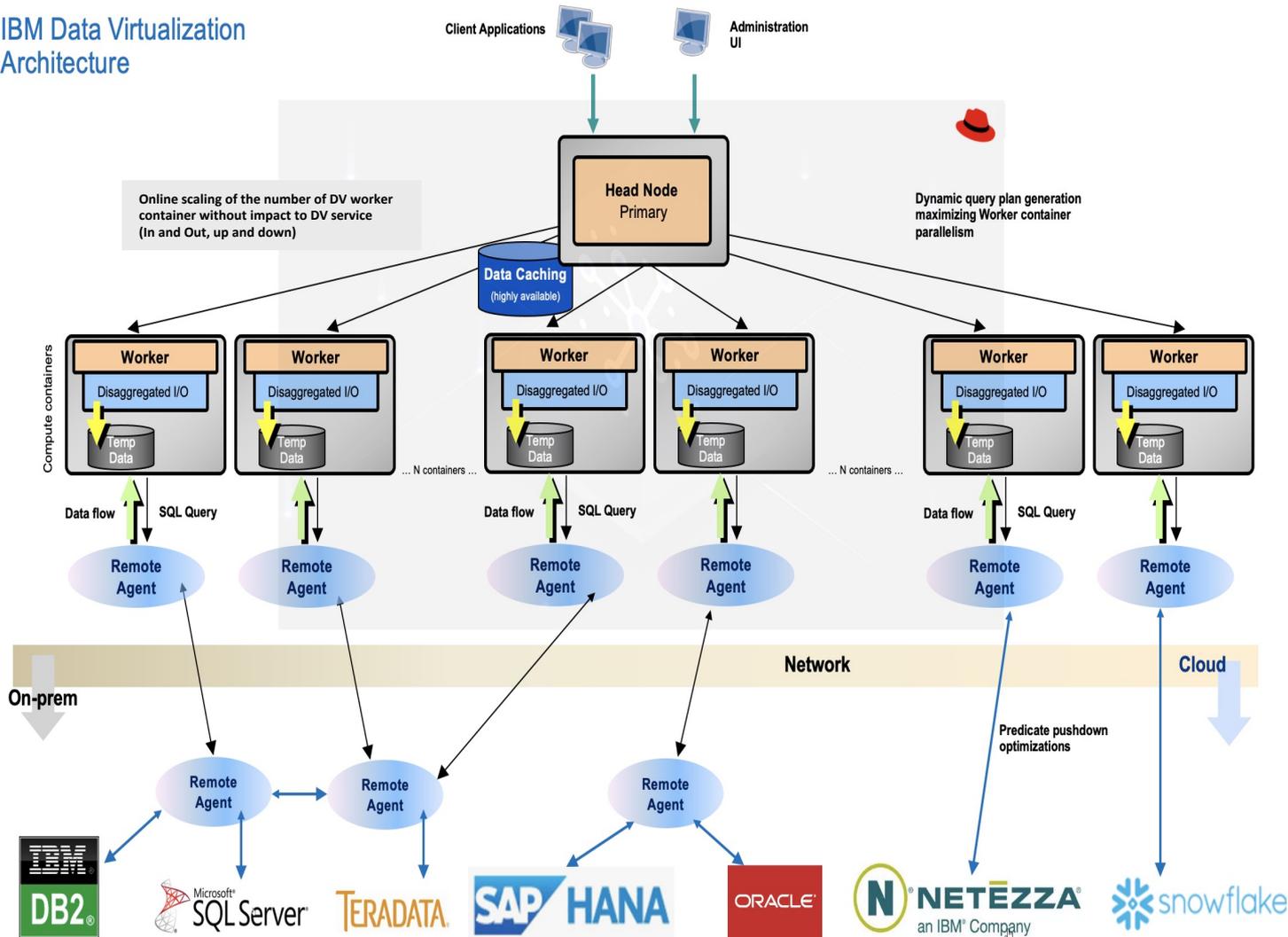
- Parallel processing mesh providing execution performance and scalability:
 - *Quickly deliver analytics results and easily evolve with new data source demands*
- Provides resilient connections between data sources:
 - *Reliability and ability to quickly adapt to increasing business demand*
- Scales seamlessly as new sources are added:
 - *New remote connectors and data sources can be added to the processing mesh without interruption of the service.*
- Provides data source discovery for data sources outside of CP4D
 - *Access to files on disk*
 - *Data sources outside of the cluster.*
- Richness of automation:
 - *Automatic formation and reorganization for best performance.*
- Highly distributed processing:
 - *Parallel access to data and query results.*
 - *Distributed algorithms to improve query performance*



Data Virtualization architecture provides multiple layers that can be tuned according to the workload.

- Multi-worker support in the cluster scales both horizontally and vertically to handle high concurrency and data volumes.
- DV remote agents scale to support higher number of connected data sources and multi-source queries.
- Optimizer dynamically chooses the best number of worker to use for portions of the query based on data sizes and workload

IBM Data Virtualization Architecture



Cloud Pak for Data

DV 1H 2022, DV aaS MVP on 1H, 2022

- Db2 family for HDM
- Db2 for iSeries, zSeries
- Db2 for z/OS
- Big SQL
- IIAS, PDA (Netezza)
- Informix
- Amazon RDS (Oracle, SQLServer)
- Denodo (Data Virtualization as a source)
- Derby
- Oracle
- SQL Server
- MySQL
- PostgreSQL
- Apache Hive, HDP Hive
- Cloudera Impala
- Teradata*
- MongoDB
- Hive
- Excel, CSV, Text*[@]
- Sybase
- MariaDB
- Snowflake
- Z Data Sources through IBM DVM Integration
 - VSAM, IMS, CICS, Adabas
- Map-R (Hive)
- BigQuery
- SAP S4/HANA & BW (JDBC / OData connection)
- Amazon Redshift
- Salesforce
- Greenplum
- Apache Spark SQL
- COS / S3

In the roadmap pipeline

- TM1 (REST API)
- Apache Kafka
- Cassandra
- SAS
- JSON
- REST API
- TIBCO (Data Virtualization as a source)
- Interbase
- Apache Drill
- Amazon Dynamo DB, Aurora DB
- CouchDB
- Stream / MQ
- Cloudbant
- DataBricks Delta Lake

**Broad support
for common
data source
types**

*More to be added
in the future.*

Key Features

Collaborative Compute Model

Remote Processing and Discovery

Governance Integration

Caching

Schema Folding

Compute Scalability

Data and Result Caching

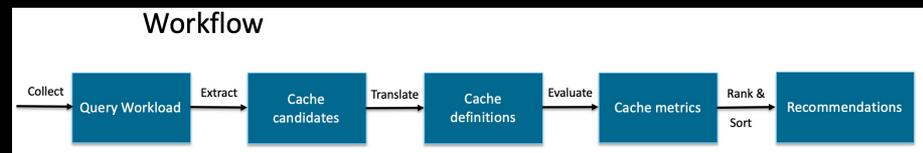
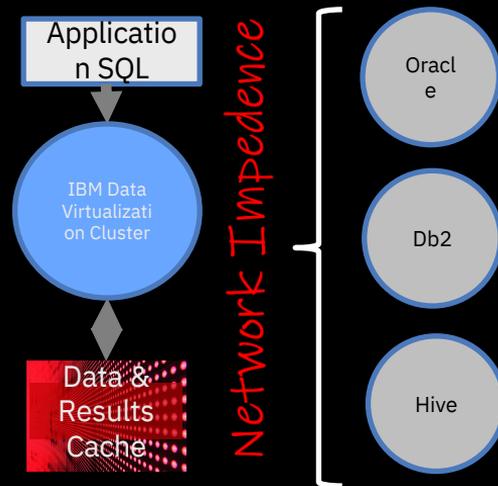
Automated cache recommendations to further optimize query performance-

Powerful

- Cache results (common SQL statements)
- Cache data (data or aggregates, etc).
- Define refresh rate
- Monitor usage/effectiveness

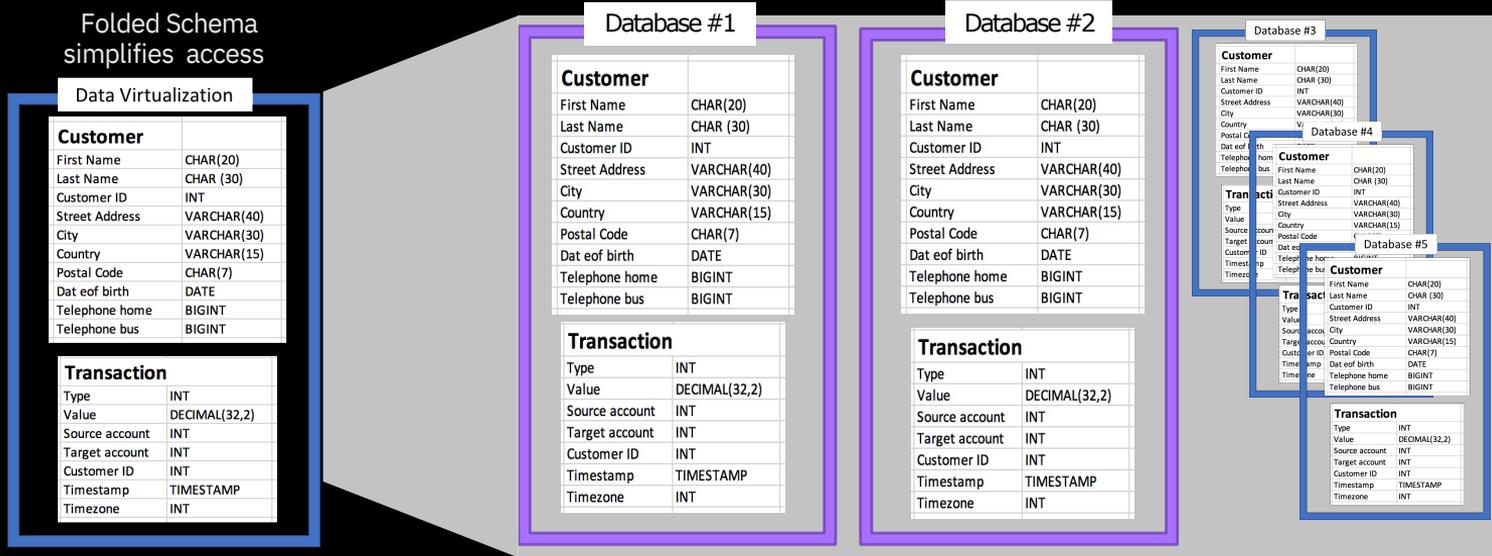
Under the hood

- Advanced query compiler determines whether to use cached data and results for part or all of a query result (MQT & ML based)
- Automation for implementing cache based on your query history



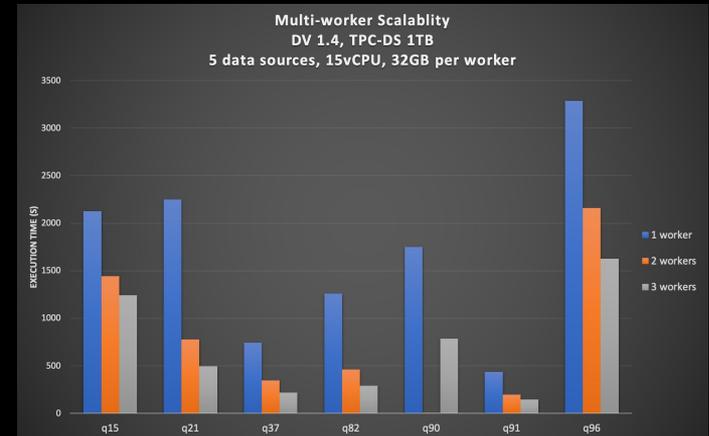
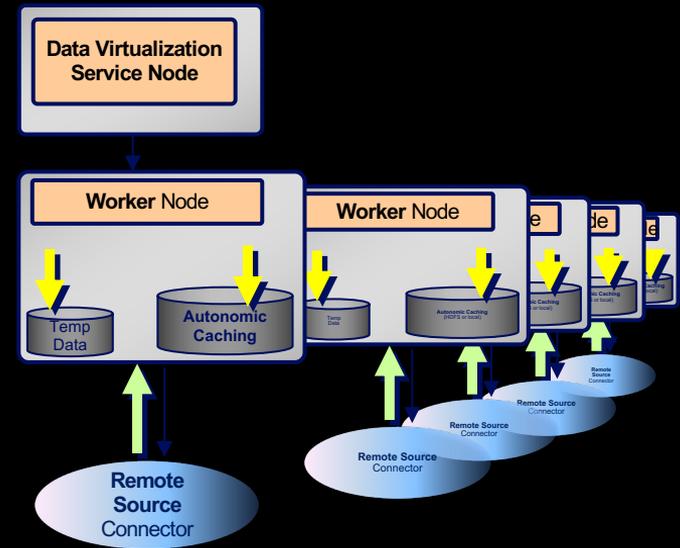
Schema Folding – Simplify Your Data

- Common or similar schemas appear in multiple databases.
 - E.g. branch database for a bank or retailer.



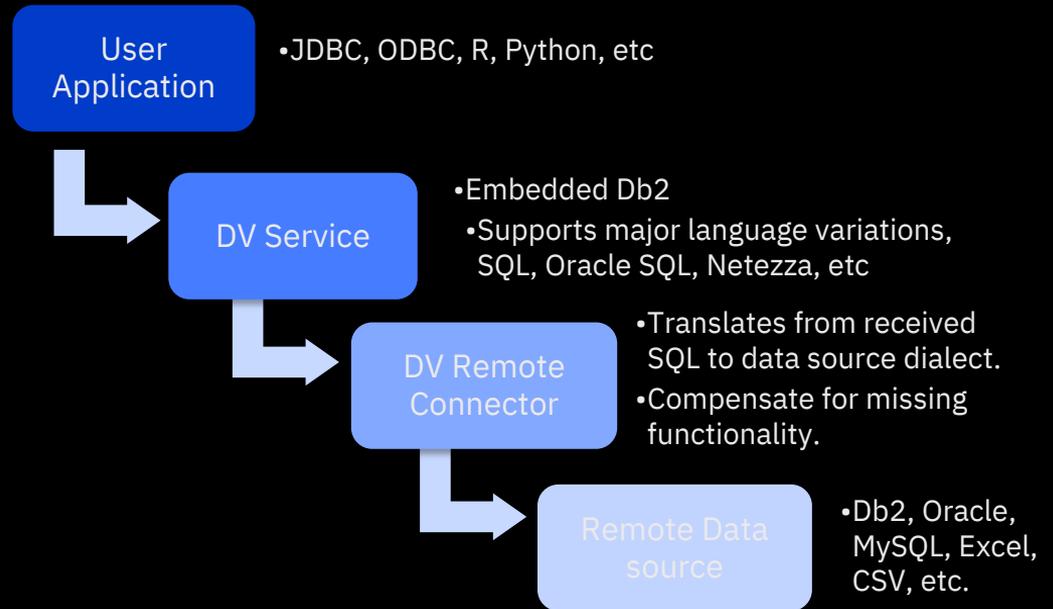
Service Compute Scalability

- Supporting multiple worker nodes to improve query performance for large workloads and data sets.
- Parallelized data fetch from data sources.
- When using remote connectors, additional advantages for network parallelism as data can flow on separate network routes to reach the cluster.



Language Translation in Data Virtualization

- Broad set of data sources supported by Data Virtualization each with unique syntax variations.
- Constellation is not limited only a single data source type. A logical schema is created across all connected sources.
- Multiple levels of translation as we move from the applications through the constellation down to the data source.



Security

Overall approach to Governance Integration and Security.

- Controlled, governed and secure access to virtual data sets within the Cloud Pak for Data Platform

Governance

- Provides data classification (business term) and policy control for virtualized data from any source.
- Data provenance and lineage to show how each source contributes to the analytics.

Security

- Data lives in the original source with only results ever being transmitted.
- Fully encrypted communications between all components of the solution.
- Strong authentication and permission management.
- Policy enforcement and masking capabilities.
- User Group Support *

Policies, via a Runtime Catalog

- Policies can capture corporate data mandates and can have sub-policies
- Additional rules can be associated with policies.
- A data protection rule can be added to the policy so that the policy to protect sensitive data can be enforced.

Data Virtualization Today

Governance Policy and rules apply across

Governance in real-time for all virtual objects

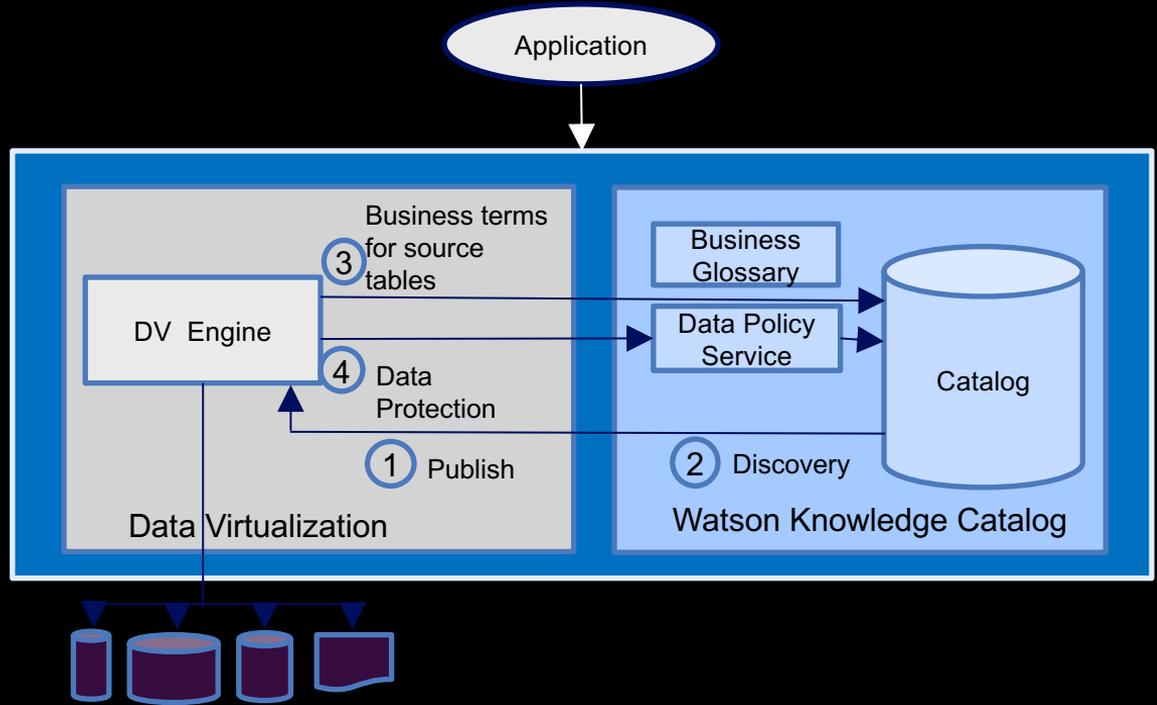
- Govern the Data fabric using policies, data protection rules, or deny policy across all Virtual objects
- Apply masking rules to your virtual objects in real-time
- With deep integration policies and rules result in same behavior when accessed within, or outside of Cloud Pak for Data platform
- Uniformity of enforcement for all virtual objects , irrespective of the Data Sources data type or structure

The image displays two screenshots of the IBM Cloud Pak for Data interface. The top screenshot shows a 'Data sources' view with a network diagram of data sources and a 'PATIENT' table structure overview. The bottom screenshot shows a detailed view of the 'PATIENT' table with 7 columns and 20 rows of data, including columns for SSN, USERID, NAME, ADDRESS, PHARMACY, ACCT_BALANCE, and PCP_ID.

SSN	USERID	NAME	ADDRESS	PHARMACY	ACCT_BALANCE	PCP_ID
XXXXXXXXXX	peter125	Peter Gerry	Oak Wood St	Kroger Pharmacy	335	314-11122
XXXXXXXXXX	kanpeng111	Kan-Peng	Cottle Circle	Booths	761	323-63541
XXXXXXXXXX	jones999	Jerry Ho	Almaden Circle	San Jose Pharmacy	2121	193-18232
XXXXXXXXXX	alan888	Ellen Smiles	Home Court	Mays	121	414-76439
XXXXXXXXXX	jmjm666	Jim Meyers	Chardonway St	Mays Pharmacy	431	124-12427
XXXXXXXXXX	matbowen	Matt Bowen	Boeing Ave	ConcencPharmacy	200	317-34484
XXXXXXXXXX	mar411	Mark Bonner	Cabrilly Drive	CVS	248	132-86878
XXXXXXXXXX	kyler502	Kyle Reid	Eden Square	Quest Pharmacy	113	422-11-35765
XXXXXXXXXX	ray125	Tim Fay	Hollywood Court	Walgreens Supercenter Pharmacy	231	514-20141
XXXXXXXXXX	qbbs001	Peter Rabbit	London Way	English Pharmaceuticals	1327	321-3838
XXXXXXXXXX	brad1	Bredley Carr	Fremont Ave	Lucky Pharmacy	1821	643-11292
XXXXXXXXXX	kim1345	Kimberly Penn	500 State St	Walgreens	123.45	123-45678
XXXXXXXXXX	chuck0045	Charles Drew	451 City Avenue	CVS Pharma	397.25	779-32754
XXXXXXXXXX	MoeBarkley123	Moe Barkley	777 Murphy Avenue	Safeway Pharmacy	245.1	443-19031
XXXXXXXXXX	lyl3000	Lily Lai	Sterling Centre Drive	Walgreens	645	410-73451
XXXXXXXXXX	emily222	Emily Cruz	Sea Breeze Court	Walmart Pharmacy	377	210-30274

Integrated Governance

- I. DV Publishes remote assets to WKC.
- II. WKC Discovery performs classification, scoring, term association
- III. DV retrieves Business Terms for the source tables to give the users a common understanding of the data.
- IV. DV obtains information about the policies it needs to enforce upon data access.



What problem are we solving? What's the opportunity?

Exposing Hybrid Data Quickly via Data Fabric (& Key Attributes)

What? Hybrid data fabric for *making existing platforms flexible*

- Use data without knowing format/location, moving or copying

Why? Constant change and need for automated governance

- Ease of query / access to query across sources
- Hybrid: Connect and Access Across Sources
- Scalable and Fast: 1 or 1,000s!
- Distributed: Regional Storage/Governance & Parallel Access
- Governed: Policies applied during search, query, usage

Benefits:

- Time/Effort to Access: from Months to Minutes
- Use of Data: Effective/Proper, in-context
- Time to Value/Innovation: 80% of time spent understanding data
- Tech / Data Modernization: technology modernization/take-out, isolating data impacts (e.g. Data Lakes)

Steps to increase Time-to-Result, with Reduced Effort

1. Connect to Existing Data Assets (DBs, etc.) or Search Cataloged Data Assets
2. If the User/SME doesn't already know the "design", ...
 - Discover and Profile the Quality of Data Assets
 - Discover the De Facto "Design" between Hybrid Sources (PK->FK Key Relationship Analysis)
3. Create a new Virtual Object, Querying Across Data Sources
 - Based on already-known de-facto "design" PK->FK Key Relationships
4. Make Virtual Data Available to Others (Users, Platforms, Applications)

1. Connect to Existing Data Assets (DBs, etc.) or Search Cataloged Data Assets

The screenshot shows the 'Data virtualization' console. On the left, there is a sidebar with a 'Filters' section and a list of databases including IBM Db2 Family (414), MongoDB (19), Informix (3), IBM BigSQL (0), IBM Db2 Event Store (0), Derby (0), z/OS data source (non-Db2z) (0), and Hive (0). The main area is titled 'Available tables' and shows 433 tables. A search bar at the top of this area contains the text 'Find tables by name, schema, or column'. Below the search bar is a table with columns: Table, Schemas, Databases, and Grouped tables. The table lists several tables, including MORTGAGE_JOIN, MORTGAGE_CUSTOMER, MORTGAGE_DEFAULT, MORTGAGE_PROPERTY, BindingDimensions, CUSTOMER, DebiasedFairnessMetrics, DeploymentDimensions, DeploymentLatestMetrics, and Explanations.

The screenshot shows the 'Data virtualization' console with the title 'My virtualized data'. A search bar at the top contains the text 'mort'. On the right side of the search bar are 'Assign' and 'Join view' buttons. Below the search bar is a table with columns: Table, Schema, and Created on. The table lists several virtualized data assets, including MORTGAGE_CUSTOMER_DB2, MORTGAGE_JOIN_DB2, MORTGAGE_RISK_PRED_DB2, MORTGAGE_PROPERTY_MONGO, mortgage_default_informix, and MORTGAGE_CUSTOMER_MONGO.

Table	Schema	Created on
<input checked="" type="checkbox"/> MORTGAGE_CUSTOMER_DB2	DB2	Dec 13, 2019 6:12:12 PM
<input type="checkbox"/> MORTGAGE_JOIN_DB2	DB2	Dec 12, 2019 9:59:59 PM
<input type="checkbox"/> MORTGAGE_RISK_PRED_DB2	DB2	Dec 12, 2019 9:59:59 PM
<input checked="" type="checkbox"/> MORTGAGE_PROPERTY_MONGO	MONGO	Dec 12, 2019 9:56:20 PM
<input type="checkbox"/> mortgage_default_informix	INFORMIX	Dec 12, 2019 9:56:19 PM
<input type="checkbox"/> MORTGAGE_CUSTOMER_MONGO	MONGO	Dec 12, 2019 9:56:19 PM

If needed, Discover and Profile the Quality of Data Assets

IBM Cloud Pak for Data

Projects > AA_Dan0629a > Data assets > BANK_CLIENTS

Columns Governance Data quality Data classes Data types Rules Primary keys Foreign keys

BANK_CLIENTS

Description
Overall Description for BC manually added
7/16/2022... Show more

Data quality score 96% 0%
Columns 17
Rows 5112
Reviewed
Threshold 80%
Analysis status Completed
Last analysis Jul 16, 2020

Find a column

Name (physical)	Score	Delta
SQL170621015...	100%	0%
CLIENT_ID	100%	0%
NAME	98%	0%
ADDRESS	88%	0%
ZIP	81%	0%
AGE	92%	0%
GENDER	100%	0%
MARITAL_STATUS	90%	0%
PROFESSION	97%	0%
NBR_YEARS_CLI	99%	0%
SAVINGS_ACCO...	100%	0%
ONLINE_ACCESS	100%	0%
JOINED_ACCO...	100%	0%
BANKCARD	100%	0%

Columns (17)

Search for a column

<input type="checkbox"/>	Name	Analysis status	Last analyzed	Data class	Term	Format	Nullability	Uniqueness	Minimum	Maximum	Distinct values
<input type="checkbox"/>	ACCOUNT_ID	Completed	28 days ago	Account Number							
<input type="checkbox"/>	ACCOUNT_TYPE	Completed	28 days ago	Indicator							
<input type="checkbox"/>	ADDRESS	Completed	28 days ago	Text	Address 100%						
<input type="checkbox"/>	AGE	Completed	28 days ago	Code	Age 100%						
<input type="checkbox"/>	AVERAGE_BALANCE	Completed	28 days ago	Quantity							
<input type="checkbox"/>	BANKCARD	Completed	28 days ago	Boolean							
<input type="checkbox"/>	CLIENT_ID	Completed	28 days ago								
<input type="checkbox"/>	GENDER	Completed	28 days ago	Gender	Gender 100%						
<input type="checkbox"/>	JOINED_ACCOUNTS	Completed	28 days ago	Boolean							
<input type="checkbox"/>	MARITAL_STATUS	Completed	28 days ago	Legal Marital/Civil Status...	Marital Stat...						
<input type="checkbox"/>	NAME	Completed	28 days ago	Person Name	Name 100%						
<input type="checkbox"/>	NBR_YEARS_CLI	Completed	28 days ago	Code							
<input type="checkbox"/>	ONLINE_ACCESS	Completed	28 days ago	Boolean							

IBM Cloud Pak for Data

Data discovery > Results

Results for qs_1590090084338

Explore assets

Discovery insights Data quality insights

Business term assignment

Category	Percentage
Assigned	4.9
Suggested	44.28
Unassigned	94.1

Data class assignment

Category	Percentage
Assigned	60.03
Suggested	26.26
Unassigned	39.97

All business terms (33)

Name	Columns	Tables	Schemas	% assigned data class	% suggested data class	Assigned data class	Suggested data class
Age	12	0	0	75%	67%	Identifier	US Zip Code
Application	12	1	0	46%	46%	Identifier	Person Name
Name	5	0	0	40%	60%	Text	Person Name

IBM Cloud Pak for Data

Projects > Austin_CS_demo

Dashboard Data assets Data rules Relationships Column similarity Settings

Austin_CS_demo

Description
No description.

Data assets 27
PII data assets 0/27
Reviewed data assets 0/27

Connections 3
Critical data issues 0

Created by admin
Last modified Jun 18, 2020

● Governed ● Sampling enabled

Data quality threshold

Category	Percentage
No quality score	18%
Conformed	44%
No threshold	37%



If needed, Discover the De Facto “Design” between Hybrid Sources

(PK-

The screenshot displays the IBM Cloud Pak for Data Relationships interface. The top navigation bar includes 'Dashboard', 'Data assets', 'Data rules', 'Relationships', 'Column similarity', and 'Settings'. The 'Relationships' section is active, showing a 'Data asset relationships chart' and a table of 'Found 5 relationships'.

Data asset relationships chart: This chart visualizes relationships between six data assets: BANK_ACCOUNTS, BANK_CLIENTS, BANK_CUSTOMERS, SAVINGS_ACCOUNTS, CHECKING_ACCOUNTS, and ACCOUNT_HOLDERS. Each asset is represented by a box containing its schema. Relationships are indicated by dashed lines connecting primary keys in one asset to foreign keys in another.

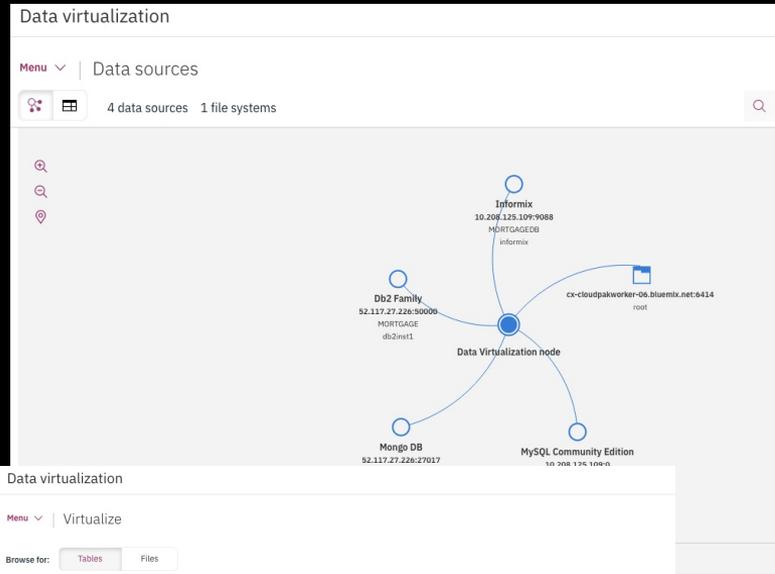
- BANK_ACCOUNTS:** ACCOUNT_BALANCE (Numeric), ACCOUNT_ID (PK, Numeric), ACCOUNT_TYPE (String), BANKCARD (String), CUSTOMER_ID (FK, Numeric), JOINT_ACCOUNT_HOLDER (String), ONLINE_ACCESS (String), SQL70921019632390 (Unknown).
- BANK_CLIENTS:** ACCOUNT_ID (PK, Numeric), ACCOUNT_TYPE (String), ADDRESS (String), AGE (Numeric), AVERAGE_BALANCE (Numeric), BANKCARD (String), CLIENT_ID (FK, String), GENDER (String), JOINED_ACCOUNTS (String), MARITAL_STATUS (String).
- BANK_CUSTOMERS:** ADDRESS (String), AGE (Numeric), CREDIT_RATING (Numeric), CUSTOMER_ID (PK, Numeric), GENDER (String), MARITAL_STATUS (String), NAME (String), NBR_YEARS_CL (Numeric), PROFESSION (String), ZIP (String).
- SAVINGS_ACCOUNTS:** ACCOUNT_BALANCE (Numeric), ACCOUNT_ID (PK, Numeric), ACCOUNT_HOLDER_ID (FK, Numeric).
- CHECKING_ACCOUNTS:** ACCOUNT_BALANCE (Numeric), ACCOUNT_ID (PK, Numeric), ACCOUNT_HOLDER_ID (FK, Numeric).
- ACCOUNT_HOLDERS:** ACCOUNT_HOLDER_ID (PK, Numeric), ADDRESS (String), AGE (Numeric), GENDER (String), MARITAL_STATUS (String).

Found 5 relationships table:

Parent data asset	Primary key	Columns	Child data asset	Foreign key	Parent to child	Child to parent	Confidence	Status	Last m	Set status	Remove relationship	
<input checked="" type="checkbox"/>	BANK_CLIENTS	ACCOUNT_ID	1	SAVINGS_ACCOUNTS	ACCOUNT_ID	43% (2213 / 5112)	100% (2213 / 2213)	71%	Candidate	6/29/2020, 10:03 AM	Selected	Rejected
<input type="checkbox"/>	BANK_ACCOUNTS	ACCOUNT_ID	1	SAVINGS_ACCOUNTS	ACCOUNT_ID	62% (2213 / 3532)	100% (2213 / 2213)	81%	Candidate	6/29/2020, 10:03 AM		
<input type="checkbox"/>	BANK_ACCOUNTS	ACCOUNT_ID	1	CHECKING_ACCOUNTS	ACCOUNT_ID	62% (2199 / 3532)	100% (2199 / 2199)	81%	Candidate	6/29/2020, 10:03 AM		
<input type="checkbox"/>	BANK_CUSTOMERS	CUSTOMER_ID	1	ACCOUNT_HOLDERS	ACCOUNT_HOLDER_ID	97% (2941 / 3008)	100% (2941 / 2941)	98%	Candidate	6/29/2020, 10:03 AM		
<input type="checkbox"/>	BANK_CLIENTS	ACCOUNT_ID	1	CHECKING_ACCOUNTS	ACCOUNT_ID	43% (2200 / 5112)	100% (2199 / 2199)	71%	Candidate	6/29/2020, 10:03 AM		

Items per page: 10 | 1-5 of 5 items

3. Create a new Virtual Object, Querying Across Data Sources (based on de-facto “design” PK->FK Key Relationships)



Data virtualization

Menu | Virtualize

Browse for: Tables Files

Find tables by name, schema, or column

Filters Available tables Automatically group tables

Databases 36 tables

Table	Schemas	Databases	Grouped tables
MORTGAGE_JOIN	DB2INST1	MORTGAGE	1
MORTGAGE_CUSTOMER	DB2INST1.ICPD_MONGO	MORTGAGE.icpd_mongo	2
MORTGAGE_DEFAULT	DB2INST1.ICPD_MONGO	MORTGAGE.icpd_mongo	2
MORTGAGE_PROPERTY	DB2INST1.ICPD_MONGO	MORTGAGE.icpd_mongo	2
sysdomains	"informix"	MORTGAGEDB	1

Join virtual objects

Join two columns by selecting a column from one table and then dragging your cursor to a column in the other table.

Table 1: CS_MORTGAGE_PROPERTY

Column Name	Data Type
ID	DOUBLE
LOCATION	INTEGER
SALE_PRICE	INTEGER

Table 2: CS_MORTGAGE_CUSTOMER

Column Name	Data Type
APPLIED_ONLINE	VARCHAR
CREDIT_CARD_DEBT	DOUBLE
ID	DOUBLE
INCOME	DOUBLE
LOANS	DOUBLE
LOAN_AMOUNT	DOUBLE

IBM Cloud Pak for Data

SQL editor

```
1 CREATE VIEW VIEW_NAME
2 AS
3 SELECT "DB2"."MORTGAGE_CUSTOMER_DB2"."APPLIED_ONLINE" AS "DB2_MORTGAGE_CUSTOMER_DB2_APPLIED_ONLINE", "DB2"."MORTGAGE_CUSTOMER_DB2"."CARD_DEBT" AS "DB2_MORTGAGE_CUSTOMER_DB2_CARD_DEBT", "I
4 FROM "DB2"."MORTGAGE_CUSTOMER_DB2"."MONGO"."MORTGAGE_PROPERTY_MONGO"
5 WHERE "DB2"."MORTGAGE_CUSTOMER_DB2"."ID"="MONGO"."MORTGAGE_PROPERTY_MONGO"."ID" AND "CURRENT_LOANS" = 1
```

4. Make Virtual Data Available to Others (Users, Platforms, Applications)

Data virtualization

Menu ▾ | Connection details

Linux Mac PowerLinux Windows zLinux

Instructions

- [Download Linux Driver Package](#)

File name:
ibm_data_server_driver_package_linuxx64.tar.gz (74 MB)
- Run the following example commands to decompress the ibm_data_server_driver_package_linuxx64.tar.gz file:

```
gunzip ibm_data_server_driver_package_linuxx64.tar.gz
tar -xvf ibm_data_server_driver_package_linuxx64.tar.gz
```

A dsdriver subdirectory is created.
- Extract the Java and ODBC/CLI drivers by running the following command from the dsdriver directory:

```
./installDSDriver
```

The **installDSDriver** command creates the db2profile and db2cshrc script files in the dsdriver directory.
- Run the script file for your shell environment:

Connection configuration resources

With SSL Without SSL

Host name: zen-cpd-zen.apps.cpd-cluster-demo2.demo.ibmcloudpack.com
Port number: 30558
Database name: bigsql
User ID: user999
Version: Compatible with Db2, Version 11.1.0 or later
Instance ID: d4b76754-64eb-47a1-9d2d-e1b1f83c0f1e

[Download SSL Certificate](#)

JDBC string

```
jdbc:db2://zen-cpd-zen.apps.cpd-cluster-demo2.demo.ibmcloudpack.com:30558/bigsql:user=user999
```

[Copy JDBC String](#)

More information

- [Db2 driver package \(IBM Knowledge Center\)](#)
- [IBM Data Server Client Packages](#)
- [Connecting CLPPlus to a Db2 database \(IBM Knowledge Center\)](#)

