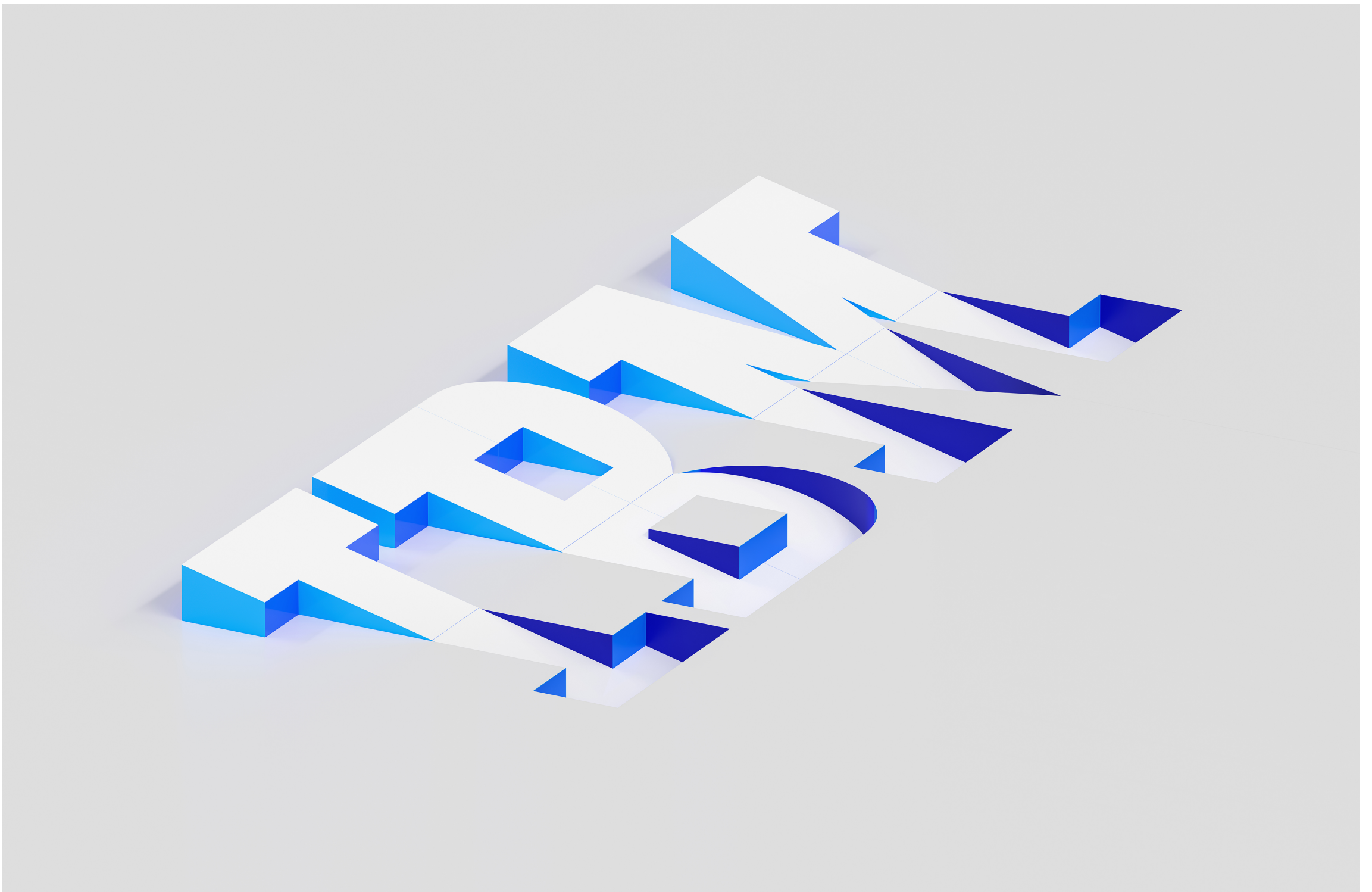


IBM Cloud

Db2 Warehouse as a Service

Product guide



Edition notices

This PDF was created on 2025-03-11 as a supplement to *Db2 Warehouse as a Service* in the IBM Cloud docs. It might not be a complete set of information or the latest version. For the latest information, see the IBM Cloud documentation at <https://cloud.ibm.com/docs/Db2whc?topic=Db2whc-getting-started>.

About

IBM® Db2® Warehouse as a Service is a fully-managed, elastic cloud data warehouse that delivers independent scaling of storage and compute. It delivers a highly optimized columnar data store, actionable compression, and in-memory processing to supercharge your analytics and machine learning workloads.

Key features

Blazingly-fast engine

IBM Db2 Warehouse SaaS is a column-organized, in-memory data warehouse designed for complex analytics and extreme concurrency. IBM Db2 Warehouse SaaS delivers high performance on complex analytics workloads by using IBM BLU Acceleration, a collection of technologies pioneered by IBM Research that features the following key optimizations:

- A columnar organized storage model
- In-memory processing
- Querying of compressed data sets
- Data skipping

Scalable and elastic cloud service

IBM Db2 Warehouse SaaS is an elastic service, and allows you to independently scale compute and storage through an intuitive UI or REST API.

Native object storage support

IBM Db2 Warehouse SaaS on Amazon Web Services (AWS) introduces native object storage support to Db2. This functionality allows you to store traditional Db2 column-organized tables in Db2's native format on object storage while maintaining the existing SQL support and performance using a tiered storage architecture. Moving data from higher cost block storage to lower cost object storage can significantly reduce your costs and does not require changes to your workload. This feature is available on the current generation of AWS plans. For more information, see [Native Cloud Object Storage support](#)

Open data format table support

IBM Db2 Warehouse SaaS on AWS introduces new DATALAKE table support. This functionality provides Db2 users with the ability to access data that is stored on inexpensive, scalable, and reliable object storage in open data formats such as PARQUET and ORC from within the Db2 Warehouse. This feature is available on the current generation of AWS plans. For more information, see [Open data format table support](#)

Replication Support

IBM Db2 Warehouse SaaS with Q Replication enables high-speed data replication for business continuity. This feature allows you to manage downtime with continuous availability (99.999%) and in-place recovery within a cluster. The cloud-native architecture of Db2 features multiple layers of resiliency with managed computation, highly available storage, and cross-cloud replication.

Expertly managed and highly secure

Day-to-day operations for IBM Db2 Warehouse SaaS, including database monitoring, uptime checks and failovers, are fully automated. Operations are supplemented by a DevOps team that are on call to handle unexpected system failures. Data is encrypted at rest and in motion by default. Administrators can also restrict access to sensitive data through data masking, row permissions, and role-based security, and can utilize database audit utilities to maintain audit trails for their data warehouse.

Highly resilient and reliable

IBM Db2 Warehouse SaaS provides double protection for your data with built-in snapshot backups and geo-replicated disaster recovery backups.

Compatible with on-premises data warehouses

IBM Db2 Warehouse SaaS can be configured with Oracle compatibility, so applications that were written for an Oracle database can run against IBM Db2 Warehouse SaaS without having to be rewritten. IBM Db2 Warehouse SaaS also comes with nearly 100% DDL, DML, and stored procedure compatibility with IBM Netezza®.

Built-in machine learning and geospatial capabilities

Train and run machine learning models directly in the IBM Db2 Warehouse SaaS engine, using SQL, Python, or R. IBM Db2 Warehouse SaaS also supports spatial analytics with Esri compatibility, including Esri data types such as GML, and integration with Jupyter Notebooks.

Plans and configurations

For information about the plans and configurations supported on IBM Cloud and Amazon Web Services (AWS), see [here](#).

This set of documentation covers the detailed commands and reference topics for the Db2 engine that powers IBM Db2 Warehouse SaaS. To find the IBM Cloud documentation for the offering, see [here](#). The IBM Cloud documentation covers the high-level functionality for the cloud offering, and refers back to this set of documentation where appropriate.

Supported cloud providers and data centers

IBM Db2 Warehouse SaaS can be deployed on both IBM Cloud and Amazon Web Services.

Availability of plans in IBM Cloud data centers

The following table provides information about the availability of the various IBM Db2 Warehouse SaaS plans by IBM Cloud data centers that are located in geographical regions:

IBM Db2 Warehouse SaaS plans	Asia/Pacific	Europe	North/Central America	South America
Flex	Tokyo	Frankfurt	Dallas (us-south)	*NA
			Washington D.C. (us-east)	
Flex Performance	Tokyo	Frankfurt	Dallas (us-south)	*NA
			Washington D.C. (us-east)	
Flex One	Tokyo	Frankfurt	Dallas (us-south)	*NA
		London	Toronto	
			Washington D.C. (us-east)	

Table 1. IBM Cloud data centers supporting IBM Db2 Warehouse SaaS service plans

*NA = Not available at this time

For information on deploying your IBM Db2 Warehouse SaaS service in a network-isolated environment on the IBM Cloud, open a [support case](#).

Availability of plans in Amazon Web Services data centers

The current generation of plans on AWS is available in the following data centers:

IBM Db2 Warehouse SaaS plans	Asia/Pacific	Europe	North/Central America	South America
All current generation	Tokyo	Frankfurt	N. Virginia	Sao Paulo
	Sydney			

Table 2. Amazon Web Services data centers supporting IBM Db2 Warehouse SaaS service plans

Support for additional AWS regions is coming soon.

The previous generation of plans on AWS is available in N. Virginia, Oregon, Frankfurt, London, Tokyo, Seoul, Singapore, and Sydney.

If you wish to provision a service instance in an AWS region not listed above, contact your local sales representative or open a [support case](#). You can also open a support case for information on deploying your IBM Db2 Warehouse SaaS service in a network-isolated environment on AWS.

Getting started with IBM Db2 Warehouse as a Service

You can provision an instance of IBM Db2 Warehouse SaaS through the [IBM Cloud catalog](#). Create a [free account](#) and get an IBM Cloud credit of \$200 that you can use towards IBM Db2 Warehouse SaaS. You can also get a \$1000 promo code to try out IBM Db2 Warehouse SaaS by following the instructions [here](#).

After creating the IBM Db2 Warehouse SaaS service, you can create a user name and password by clicking the **Service credentials** tab on your service page and selecting **New credential**.

While logged in as the **IAM** user that provisioned the instance, you can log into the web console by clicking on the **Go to UI** button on the **Manage** tab. In order to activate and administer Q Replication, you must login as a **IAM** user with **JDBC** privileges.

You can create database/JDBC users, which you will use to connect to the database, by navigating to the **Administration --> User Management** panel. Additionally, you may want to give admin access to an IAM user to allow them to use console administration functions such as user management, backup, and restore. Before you can add an IAM user in the console, you must first give the user access to the service in IAM". For more information, see [Identity and access management \(IAM\) on IBM Cloud](#).

Connecting

Connecting overview

You can connect command-line interfaces, IBM® or third-party applications and tools, or apps that you create to your IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the prerequisites.

- Collect database details and credentials

To connect to your database, you need database details (such as the host name) and credentials (such as a user ID and password.) You can collect this connection information from the IBM Db2 Warehouse SaaS web console.

- Verify that a supported driver is installed

- If your application or tool already contains the Db2 v11.1 IBM Data Server Driver Package, then your application or tool is able to connect to your IBM Db2 Warehouse SaaS database by using that driver.
- Otherwise, install the Db2 driver package, which you can download from the IBM Db2 Warehouse SaaS web console.

- Configure your environment

- Add entries to the driver configuration file, `db2dsdriver.cfg`, for your database.
- Secure Sockets Layer (SSL)

Connection details, such as which port to use and the connection string, depend on whether you use SSL connections. Using SSL is strongly recommended because of the stronger security it provides.

To use SSL connections, you need a CA certificate:

- If you use the most recent IBM Db2 Warehouse SaaS driver package, the certificate file is bundled with the package and used for connections.
- If you use the IBM Data Server Driver Package, you can download the SSL certificate from the IBM Db2 Warehouse SaaS web console.

- Confirm that ports are available

If your network is behind a firewall, confirm that communications are permitted on port number `50000` for standard protocols or port number `50001` for SSL connections.

Collecting connection information

- [Database details and connection credentials](#)

Downloading and installing driver package

- [Download driver package](#)
- [Installing on Linux or PowerLinux](#)
- [Installing on Mac OS X](#)
- [Installing on Windows](#)

Configuring your environment

- [Configuring your local environment](#)
- [SSL connectivity](#)

Connectivity options

IBM Db2 Warehouse SaaS offers multiple secure connectivity options depending on your application connection requirements.

- See [Connectivity options on IBM Cloud](#).
- See [Connectivity options on Amazon Web Services](#).

Connecting programmatically

You can use common programming languages to create applications that connect to a IBM Db2 Warehouse SaaS database.

- [JDBC](#)

- [Microsoft Windows ODBC or CLI](#)
- [.NET](#)
- [ODBC Data Source Administrator](#)
- [PHP](#)
- [REST API](#)

Integrating apps and tools

You can also connect external applications and tools to IBM Db2 Warehouse SaaS and use them to further manage or analyze your data. For example:

Data integration

- Connect your IBM Cloud® applications that need an analytics database.
- [DataStage](#)
- [Informatica](#)
- [Lift](#)
- [InfoSphere Data Replication](#)
- [Segment](#)
- [Data Studio](#)
- [Data Server Manager](#)
- [CLPPLUS](#)
- [Aginity Workbench to migrate Netezza® data models and data to IBM Db2 Warehouse SaaS](#)
- [InfoSphere Data Architect to design and deploy your database schema](#)

Data visualization/BI

- [Cognos Analytics to run Business Intelligence reports against your data](#)
- [Looker](#)
- [Tableau](#)
- [Microsoft Excel](#)
- [Esri ArcGIS for Desktop to perform geospatial analytics and map publishing with your data](#)

Data science

- [Watson Studio \(formerly IBM Data Science Experience\)](#)
- [SPSS Statistics](#)
- [SAS](#)
- [Local R development environment](#)

Connecting to another Db2 database

Db2 data virtualization (also known as federation) is supported by IBM Db2 Warehouse SaaS. Data virtualization gives you single-query access to all of your data that is on multiple distributed databases anywhere in your organization. You can access data that is on any of your Db2 or Informix data sources, both in the cloud and on premises.

This function is supported on all versions of IBM Db2 Warehouse SaaS.

- [Data virtualization \(federation\)](#)

Connectivity options on IBM Cloud

IBM® Db2® Warehouse as a Service offers multiple secure connectivity options for your application connection requirements.

Connecting to a public endpoint (default option)

As with any public cloud service, you can connect your application by way of a public host name that is provided to you at the time that your service is provisioned. Access to your data is protected by strong authentication, vast Db2 authorization options and access controls, encryption over the wire and at rest, and IBM security and compliance practices for development and operations. Optional IP allowlisting is offered. Create an IBM Support case if you want to enable IP allowlisting.

Important: For application connections, do not use IP addresses to connect to the IBM Db2 Warehouse SaaS instance, as the IP addresses resolved from the hostname may change.

How to connect to a public endpoint:

You can also obtain your host name and credentials in the following way:

1. Log in to IBM Cloud and click your service instance.
2. Click **Service credentials**.
3. Click **New credential**, then click **Add**.
4. After the credentials are created, under the **Actions** column, click **View credentials**.
5. In the following JSON document example, note the contents of the hostname, password, and username fields. You use these three components to make the public endpoint connection:

```
$ {
  "apikey": "abcdefghijklmnopqrstuvwxyz0123456789",
  "db": "BLUDB",
  "host": "db2w-abcdefg.eu-de.db2w.cloud.ibm.com",
  "hostname": "db2w-abcdefg.eu-de.db2w.cloud.ibm.com",
  "https_url": "https://db2w-abcdefg.eu-de.db2w.cloud.ibm.com",
  "iam_apikey_description": "Auto-generated for key crn:v1:bluemix:public:dashdb:eu-de:a/abc62e1447e5587cfcff971d4aa7d473:c1cac901-755b-489c-a742-f41295cb5dd8:resource-key:11f5e7e5-4759-439e-8291-7febc09382ce",
  "iam_apikey_name": "Service credentials-1",
  "iam_role_crn": "crn:v1:bluemix:public:iam::::serviceRole:Manager",
  "iam_serviceid_crn": "crn:v1:bluemix:public:iam-identity::a/abc62e1447e5587cfcff971d4aa7d473::serviceid:ServiceId-aecb72c2-b048-4800-a0d3-263d7bfe4e6a",
  "parameters": {
    "role_crn": "crn:v1:bluemix:public:iam::::serviceRole:Manager",
    "serviceid_crn": "crn:v1:bluemix:public:iam-identity::a/abc62e1447e5587cfcff971d4aa7d473::serviceid:ServiceId-aecb72c2-b048-4800-a0d3-263d7bfe4e6a"
  },
  "password": "hereisthepassword123",
  "port": 50001,
  "ssldsn": "DATABASE=BLUDB;HOSTNAME=db2w.abcdefg.eu-de.db2w.cloud.ibm.com;PORT=50001;PROTOCOL=TCPIP;UID=bluadmin;PWD=hereisthepassword123;Security=SSL;",
  "ssljdbcurl": "jdbc:db2://db2w-abcdefg.eu-de.db2w.cloud.ibm.com:50001/BLUDB:sslConnection=true;",
  "uri": "db2://bluadmin:hereisthepassword123@db2w-abcdefg.eu-de.db2w.cloud.ibm.com:50001/BLUDB?ssl=true;",
  "username": "bluadmin"
}
```



Figure 1. Public network access to IBM Cloud

Connecting to a private endpoint: IBM Cloud service endpoint

IBM Db2 Warehouse SaaS supports private connectivity through an [IBM Cloud service endpoint](#). IBM Cloud service endpoints securely route network traffic between different IBM Cloud services through the IBM Cloud private backplane network. When you configure your IBM Db2 Warehouse SaaS instance with IBM Cloud service endpoint connectivity, traffic between your cloud data warehouse and applications deployed on your IBM Cloud account will not traverse any public networks.

Important: For application connections, do not use IP addresses to connect to the IBM Db2 Warehouse SaaS instance, as the IP addresses resolved from the hostname may change.

How to configure IBM Cloud service endpoint connectivity

Complete the following steps to enable IBM Cloud service endpoint connectivity for your IBM Db2 Warehouse SaaS instance:

1. Enable your IBM Cloud account to use virtual routing and forwarding (VRF) and IBM Cloud service endpoints. To enable both of these items, see [Enabling VRF and service endpoints](#).
2. Configure your IBM Db2 Warehouse SaaS instance for service endpoint connectivity.
 - **If you provisioned your IBM Db2 Warehouse SaaS instance through the IBM Cloud catalog :** [Create a case](#) to request the configuration of your IBM Db2 Warehouse SaaS instance for IBM Cloud service endpoint connectivity. After this is complete, your IBM Db2 Warehouse SaaS instance will be served on a new, non-internet-routable IP address. Information about how to access your IBM Db2 Warehouse SaaS instance by using this newly configured private endpoint will be sent to you.
 - **If you purchased your IBM Db2 Warehouse SaaS instance through IBM Sales :** If you requested private endpoint connectivity, your IBM Db2 Warehouse SaaS instance will be provisioned with IBM Cloud service endpoint connectivity. No further action is required.



Note: After you've configured IBM Cloud service endpoint connectivity for your IBM Db2 Warehouse SaaS instance, it will only be accessible through a private endpoint. You will not be able to access your instance through a public endpoint.

To learn more about the IBM Cloud service endpoint service, see [Secure access to services using service endpoints](#).

Connecting to a virtual private network (VPN) endpoint

If you have an application that is deployed on a private network that is outside of the IBM Cloud without access to the public internet and you want to connect it to your database over a virtual private network (VPN) connection, you can make the request at the time that you order the service or by opening an IBM Support case. IBM network engineers will assist your network engineers to set up the VPN tunnel between your private network and the IBM Cloud.

How to connect to a VPN endpoint

To establish a VPN connection to your cloud data warehouse behind a public endpoint, [create an IBM Cloud Support case](#) that includes the following details:

- **Type of support:** Technical
- **Category:** Databases
- **Offering:** select your IBM Db2 Warehouse SaaS instance
- **Subject:** VPN Connection Request
- **Description:** provide the following required information
 - **Customer-side VPN Peer Address** (your VPN endpoint):
 - **Customer-side Encryption Domain** (be specific about what is required – 10.0.0.0/8 is unworkable because 10 addressing is also used within the IBM Cloud for back-end services):
 - **Customer-side VPN Hardware & Version :**
 - **Customer-side VPN Contact** (technical contact name and email address):
 -
 -
 -

Optional (change only if the following default values are not suitable):

IKE/ISAKMP Parameters (Phase I)

- **Encryption Method:** IKEv1
- **IKE Encryption / Encryption Algorithm:** AES-256
- **Authentication Algorithm:** SHA1
- **DH-Group:** Group 5
- **Security Association Lifetime (seconds) :** 1d (86400 seconds)

IPSec Parameters (Phase II)

- **IPSec Encryption / Encryption Algorithm :** AES-256
- **Authentication Algorithm:** SHA1
- **DH-Group** (if using PF-Secrecy): Group 5
- **Security Association Lifetime (seconds) :** 3600 seconds

After receipt of your request, IBM Cloud technicians will open the appropriate firewall ports and allowlist the provided IP address. Communication and resolution to the request is made through the IBM Cloud Support case ticket.

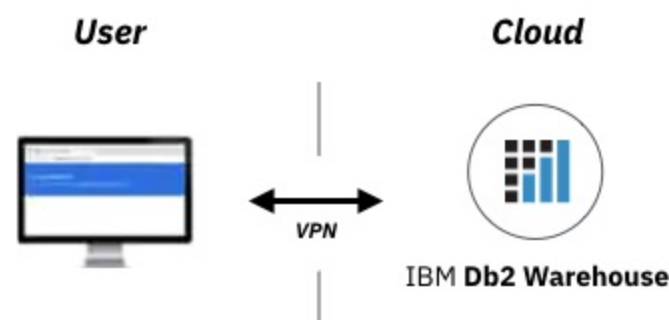


Figure 2. Public network access to IBM Cloud through a VPN

Connecting to Db2 Warehouse on Cloud with Private Link

IBM Cloud private link gives you the ability to securely and privately connect to a IBM® Db2® Warehouse as a Service instance from your own IBM Cloud VPCs. With the IBM Cloud Private Link, traffic between Db2 Warehouse on Cloud and your IBM Cloud VPCs, it does not traverse the public internet.

Complete the following steps to connect IBM® Db2® Warehouse as a Service with private link:

1. Create an access to Db2 Warehouse on Cloud console. The Db2 Warehouse on Cloud console can be accessed with IAM users, or IAM roles.
2. In the console, navigate to the **Settings** --> **Access restriction** panel then enable private endpoints. You can optionally also disable public endpoints.
3. Navigate to the **Connections** tab in Db2 Warehouse on Cloud console to get private endpoint service name and its details. Connect to DB using the connection details present in the connection's private tab after creating 'Virtual Private Endpoint Gateway'.
4. Create 'Virtual Private Endpoint Gateway', navigate to **VPC Infrastructure** -> **Virtual Private Endpoint Gateway** then click on **Create**:
 - Region - select the same region in which the VPC is created
 - Name - Unique name for the VPE gateway
 - Virtual private cloud - The VPC in which the gateway is to be created
 - Cloud service offerings - "VPE Cloud service Offering" from console Connections tab.
 - Ensure that TCP traffic is allowed through port on the VPC

Once the [VPE-endpoint-gateway](#) is created successfully, you can connect to the Db2 database using the private connections details given in the Db2 Warehouse on Cloud console.

Consideration and Limitations

- You must create the Virtual private endpoint gateway for accessing Db2 Warehouse on Cloud thru private connectivity.
- The details required for creating Virtual private endpoint gateway will be available on private connections details present in the Db2 Warehouse on Cloud console.
- You must create the Virtual private endpoint gateway in the same region where the Db2 Warehouse on Cloud instance is deployed. To access your instance from other regions, you may have to setup VPN network.

Connectivity options on Amazon Web Services

IBM® Db2® Warehouse as a Service offers secure connectivity options for your application connection requirements.

Important: For application connections, do not use IP addresses to connect to the IBM Db2 Warehouse SaaS instance, as the IP addresses resolved from the hostname may change. Use hostnames to reference your connection properties where it is available.

Connecting to Db2 Warehouse SaaS with Amazon Web Services PrivateLink

[Amazon Web Services \(AWS\) PrivateLink](#) gives you the ability to securely and privately connect to a IBM Db2 Warehouse SaaS instance that is deployed on AWS from your own AWS VPCs, services, and applications. With AWS PrivateLink, traffic between IBM Db2 Warehouse SaaS and your AWS VPCs, services, and applications does not traverse the public internet.

If you'd like to use AWS PrivateLink with IBM Db2 Warehouse SaaS, complete the following steps:

1. Create an AWS principal to access IBM Db2 Warehouse SaaS. The AWS principal can be AWS accounts, IAM users, or IAM roles.

2. Open a support ticket with IBM Cloud to enable AWS PrivateLink, and provide the Amazon Resource Name (ARN) of the AWS principal that was created in the previous step. The principal is granted permission to access your IBM Db2 Warehouse SaaS instance.
3. After the principal is granted permission, create an interface endpoint on your VPC to connect to the IBM Db2 Warehouse SaaS service. See [Creating an Interface Endpoint](#). Ensure that TCP traffic is allowed through ports 50001, 443, and 8443 on the VPC, and set rules to allow traffic from the CIDR range associated with the VPC.

Connecting to the private web console of IBM Db2 Warehouse SaaS

- When you enable private endpoint, the private web console becomes available. When private endpoint is enabled, you can still access a lite version of the web console to get connection information.
- Should you want to use the private web console, use the same AWS endpoint created from the above steps, but ensure that TCP traffic is allowed through port 8443 on your VPC.

Considerations and limitations

- AWS PrivateLink currently supports only TCP traffic. Tools that rely on UDP traffic are not supported by PrivateLink. To load data, load directly from Amazon S3 into IBM Db2 Warehouse SaaS. See [Loading data from Amazon S3](#).



Note: Extra charges might apply when you transfer data by using the public endpoint.

- You must create the Endpoint Service for accessing IBM Db2 Warehouse SaaS in the same AWS region where the IBM Db2 Warehouse SaaS instance is deployed. To access your instance from other AWS regions, you can use VPC Peering. See [Example: Services Using AWS PrivateLink and VPC Peering](#).
- For the current generation of plans on AWS, connectivity to the web UI is available only over the public network, even if you have enabled PrivateLink. This restriction is temporary, and will be removed in an upcoming update.

For more information about AWS PrivateLink, see [Interface VPC Endpoints \(AWS PrivateLink\)](#).

Database details and connection credentials

To configure your local development environment and to connect applications and tools to your IBM Db2 Warehouse SaaS database, you need to know the database details and connection credentials.

Database details

There are important database details that are needed for connecting applications and tools to your database:

- *Host name* - The host name of the server.
- *Port number* - Used by the database manager for TCP/IP communication. (Note that your service is provisioned with one port number for connections that use Secure Socket Layer (SSL) and a different port for non-SSL connection.)

If needed, you can get the IP address of your server by using the ping command or the nslookup command, specifying your host name.

- *Database name* - The Db2 database name, usually BLUDB.

Connection credentials

There are three types of credentials:

- *IBMid* - If you use IBM Cloud, this is the ID you would use to log in to IBM Cloud. This is not what you use to connect applications or tools to your Db2 Warehouse on Cloud database.
- *Db2 database credentials* - Your service comes provisioned with a database user ID and password that can be used to connect applications and tools to your database.
- *Administrator-created users* - Some IBM Db2 Warehouse SaaS plans allow administrative users to create new users. These administrator-created user IDs and passwords can be used to log directly into the web console URL and to connect to the Db2 database from applications or tools.

Where to find database details and connection credentials

You can collect this information from the following places:

- *Your administrator* - If you are not the owner or administrator of your Db2 instance, you can get your database details and connect credentials from your administrator.

- *Db2 web console* - Database details and credentials are available in the web console.
- If you are using IBM Cloud:
 - *IBM Cloud Dashboard* - When you view your service on your IBM Cloud dashboard, in the "Service credentials" area, you can view the database details as well as the database user ID and password.
 - `VCAP_SERVICES` - `VCAP_SERVICES` is an environment variable in IBM Cloud that contains database details and credentials in JSON format. When you view the service credentials for your service in your IBM Cloud dashboard, the contents of `VCAP_SERVICES` is what is displayed. When you bind other IBM Cloud services or apps to your service, those other services or apps can access database details and credentials programmatically by reading `VCAP_SERVICES`.

Dedicated private cloud for instances on Amazon Web Services

IBM® Db2® Warehouse as a Service can now be deployed on an Amazon Web Services instance in a dedicated virtual network managed by IBM.

One quantity of this part supports up to ten IBM Db2 Warehouse SaaS service instances. When a dedicated virtual network is in place, only specific AWS VPC's can access the database.

If you'd like to request this option, open a support case with IBM Cloud support to make the request. You will need to provide your IBM Cloud account ID in the ticket.

The lead time required is approximately 3-4 weeks. After your dedicated private cloud environment has been pre-built, you can deploy one or more instances into the private cloud environment by setting "Dedicated Private Cloud Configuration" to Yes when you order the instance from the IBM Cloud catalog page.

SSL connectivity

The IBM Db2 Warehouse SaaS database uses a certificate for SSL connections that is issued by a third-party digital certificate authority (CA).

The CA certificate is part of the Db2 driver package. If your application connects with a driver from the Db2 driver package, you do not need to download the certificate separately. You can download the Db2 driver package from the web console.

However, if your application has its own driver, you might need to download the certificate separately. You can download the certificate from the web console.

Secure Sockets Layer (SSL) is a security protocol that provides communication privacy. SSL enables client and server applications to communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery. SSL-enabled client applications use standard encryption techniques to help ensure secure communication.

Configuring your applications to connect to your IBM Db2 Warehouse SaaS database with SSL is strongly recommended. You should use only non-SSL connections if you are using legacy applications that cannot connect using an SSL connection.



Important: SSL connections to IBM Db2 Warehouse SaaS are enforced by default on all new and recently deployed instances and will start to be enforced on all older instances in April and May, 2020. To enable a non-SSL port on your IBM Cloud system or to keep a non-SSL port enabled, open a [support case](#) to make that request.

Configuring your Db2 client

1. [Download the IBM Global Security Kit \(GSKit\)](#) by selecting the GSKit appropriate for your operating system (OS).
2. Download the SSL certificate from the **Connection configuration resources** section under **Connections** in the **Administration** page of the IBM Db2 Warehouse SaaS web console. Store the SSL certificate file in a directory that can be referenced in a subsequent command.
3. Install the GSKit. See the following links for instructions:
 - [AIX](#)
 - [LINUX](#)
 - [WINDOWS](#)


For more details, see [IBM Global Security Kit global installation instructions overview](#).

4. Set environment variable paths:
 - AIX: `LIBPATH /usr/opt/ibm/gsk8/Lib`

- Linux: `LD_LIBRARY_PATH` `/usr/local/ibm/gsk8/lib`
- UNIX: `LD_LIBRARY_PATH` `/opt/ibm/gsk8/lib`
- Windows: `PATH` `<installation_directory>\gsk8\bin` `<installation_directory>\gsk8\lib` (`lib64` for GSKit 64-bit)

5. Create keystore. The following example command pertains to Windows:

```
gsk8capicmd_64 -keydb -create -db "mykeystore.kdb" -pw "password" -stash
```


 **Note:** You must have the ability to write to the directory or you will get an error.

6. Add SSL certificate to the keystore. The following example command pertains to Windows:

```
gsk8capicmd_64 -cert -add -db "mykeystore.kdb" -pw "password" -label ACIBLUDB_SSL -file c:\ssl\ACI_DigiCertGlobalRootCA.crt
```

7. Update the Db2 database manager. The following example command pertains to Windows:

```
db2 update dbm cfg using SSL_CLNT_KEYDB c:\PROGRA~1\IBM\gsk8\mykeystore.kdb
```

 **Note:** On Windows, `Program Files` must use `PROGRA~1`.

Connecting to your database

1. [Optional] If you use Data Studio, you can now connect to the database by selecting port `50001` and `sslConnection=true`.
2. Catalog the node and database. The following example commands pertain to Windows:

```
db2 catalog tcpip node ACICLD_S remote <IP_address_of_BLUDB_database_server> server 50001 security SSL
```

```
db2 catalog db BLUDB as ACIBLU_S at node ACICLD_S
```

3. Connect to your database with an SSL connection. The following example commands pertain to Windows:

```
db2 terminate
```

```
db2 connect to ACIBLU_S user <user_name>
```

For more information, see [Configuring Secure Sockets Layer \(SSL\) support in non-Java Db2 clients](#).

IBM Db2 Warehouse as a Service driver package

The IBM Db2 Warehouse SaaS driver package contains software for connecting client applications to a IBM Db2 Warehouse SaaS database.

About

- The driver package contains client interface tools, such as CLPPlus.
- The driver package also contains the following drivers:
 - JDBC
 - Node.js
 - Ruby
 - ODBC
 - CLI
 - .Net
 - OLE DB
 - And more ...

Already installed?

To verify that the driver package is already on your computer so that you can skip installing it again, or to determine the driver package version number, you can use the [db2level](#) command.

Downloading

You can download the driver package for your operating system from the driver download center. Your web console provides a link to the site. In your web console, select **Administration > Connections**. Select the tab for your operating system.

Installing

Install the driver package for your operating system:

- [Installing on Linux or PowerLinux](#)
- [Installing on Mac OS X](#)
- [Installing on Windows](#)

Installing the driver package on Linux or PowerLinux

You can install the IBM Db2 Warehouse SaaS driver package on Linux or PowerLinux by using `installDSDriver`.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

On PowerLinux only, complete the following steps to install the XL C/C++ compiler runtime package:

1. Download the XL C/C++ compiler runtime package from the FTP site. For example, to use the **wget** tool to download the runtime package for Linux little endian Ubuntu 14, issue the following command:

```
wget ftp://public.dhe.ibm.com/software/server/POWER/Linux/rte/xlcpp/le/ubuntu/dists/trusty/main/binary-ppc64el/*
```

2. Install the runtime package by issuing the following command:

```
sudo dpkg -iG *.deb
```

Procedure

1. Decompress the compressed driver package file that you downloaded earlier.

Example:

```
gunzip file_name.tar.gz
```

```
tar -xvf file_name.tar
```

A `dsdriver` subdirectory is created in the directory where you ran the decompression commands.

2. Extract the Java and ODBC/CLI drivers.

a. In the `dsdriver` subdirectory, run the `installDSDriver` command.

The `installDSDriver` command creates the `db2profile` and `db2cshrc` script files in the `dsdriver` directory.

b. Run one of the following script files based on your shell environment:

◦ **Bash or Korn shell:** `source db2profile`

◦ **C shell:** `source db2cshrc`

What's next?

To be able to connect your local applications or client tools to your IBM Db2 Warehouse SaaS database, [configure your local environment](#).

Installing the driver package on Mac OS X

You can install the IBM Db2 Warehouse SaaS driver package on Mac OS X by using the `installDSDriver.sh` script.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

- **For a new installation**

1. Mount the disk image by double-clicking the `macos_dsdriver.dmg` file.

A new **Finder** window opens with the contents of the disk image.

If the **Finder** window does not open, double-click the `macos_dsdriver` icon on your desktop. 2. In the **Finder** window, double-click the `installDSDriver.sh` file.

The driver package is installed in the default location: `/Applications/dsdriver`.

- **For updates to your existing driver package installation**

1. Back up current configuration files:

- a. Go the `Applications/dsdriver/cfg` folder.

- b. Copy the following files to a different folder:

```
$ `db2cli.ini`  
  
`db2dsdriver.cfg`
```

2. Remove the currently installed driver package by right-clicking the `dsdriver` folder and selecting **Move to Trash**.

3. Install the new driver package as described in the preceding **For a new installation** section:

- a. Mount the disk image by double-clicking the `macos_dsdriver.dmg` file. b. In the **Finder** window, double-click the `installDSDriver.sh` file. 4. Restore the configuration files:

Copy the `db2cli.ini` and `db2dsdriver.cfg` files that you saved from Step 1 to the `/Applications/dsdriver/cfg` folder.

What's next?

To be able to connect your local applications or client tools to your IBM Db2 Warehouse SaaS database, [configure your local environment](#).

Installing the driver package on Windows

You can install the IBM Db2 Warehouse SaaS driver package on Windows by using the installer.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. Run the downloaded executable file as an administrator.

The default installation path of the driver package is: `Program Files\IBM\IBM DATA SERVER DRIVER`

2. (Optional): Add the `bin` subdirectory of the driver package installation directory to your `%PATH%` environment variable (so you can run the `db2cli` command without specifying the full path to the command executable.)

What's next?

To be able to connect your local applications or client tools to your IBM Db2 Warehouse SaaS database, [configure your local environment](#).

Configuring your local environment

To connect local applications and tools to your IBM Db2 Warehouse SaaS database, you need to configure your environment.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. Add entries to the driver configuration file, `db2dsdriver.cfg`, for your database.

The configuration steps are different depending on whether you want to connect to your database by using SSL:

With SSL

To connect your applications and tools to your database by using SSL, enter the following commands in a command shell on Linux operating systems, at the Windows command prompt, or in a DB2 command window:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50001
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50001
```

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50001 -parameter "SecurityTransportMode=SSL"
```

where:

- `<hostname>` is the host name of your server.
- `<alias>` is an alias that you choose. The alias cannot be the same as the database name, `BLUDB`. If you want to have spaces in the alias, surround the alias with double quotation marks.

Without SSL

To connect your applications and tools to your database without using SSL, enter the following commands in a command shell on Linux operating systems, at the Windows command prompt, or in a DB2 command window:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50000
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50000
```

where:

- `<hostname>` is the host name of your server.
- `<alias>` is an alias that you choose. The alias cannot be the same as the database name, `BLUDB`. If you want to have spaces in the alias, surround the alias with double quotation marks.

2. Test connecting by issuing the **db2cli validate** command from the command prompt:

```
db2cli validate -dsn <alias> -connect -user <userid> -passwd <password>
```

where:

- `<alias>` is an alias that you created with the **db2cli writecfg** command.
- `<userid>` is your Db2 user ID.
- `<password>` is your Db2 password.

3. (Optional): To be able to connect local ODBC applications and tools to your database, register the DSN with the ODBC driver manager:

Run the following command from a command line:

```
db2cli registerdsn -add -dsn <alias>
```

where:

- `<alias>` is an alias that you created with the **db2cli writecfg** command.

By default, the DSN is created as a user DSN.

Connecting programmatically by using the driver package

The IBM Db2 Warehouse SaaS driver package contains software for connecting client applications to a IBM Db2 Warehouse SaaS database.

JDBC

Define a connection between a Java™ application and the IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

In each Java application, specify the user ID and password by including the **DriverManager.getConnection** method, and then include one of the following

JDBC URL strings:

- For a connection with SSL:

```
jdbc:db2://<host_name>:50001/BLUDB:sslConnection=true;
```

- For a connection without SSL:

```
jdbc:db2://<host_name>:50000/BLUDB
```

.NET

Define a connection between a .NET application and your IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

The following steps show you how to connect your application to the database with .NET.

1. From a command prompt, enter the following commands. These commands create new entries in the driver configuration file (`db2dsdriver.cfg`) on your computer and set the connection attributes. You need to do this step only one time.

- For a connection with SSL:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50001
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50001
```

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50001 -parameter "SecurityTransportMode=SSL"
```

where:

`<hostname>`: The host name of the server.

`<alias>`: The name for a DSN alias that you want to use to establish the .NET connection. Choose a name that is meaningful to you; for example, `analytics`.

- For a connection without SSL:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50000
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50000
```

2. (Optional): To verify the .NET connection to the database, enter the following command at a command prompt:

```
testconn40 "DATABASE=<alias>;UID=<user_id>;PWD=<password>;"
```

where:

`<alias>`: The name of the DSN alias that you created with the `db2cli writecfg` command in step 1.

`<user_id>`: Your IBM Db2 Warehouse SaaS user ID.

`<password>`: The password that you use to connect to your IBM Db2 Warehouse SaaS database.

Example

The following syntax shows sample C# code that uses the .NET driver to make a connection to the database.

```
using System;
using IBM.Data.DB2;

namespace dotNetSSLTest
{
    class Program
    {
        static void Main(string[] args)
        {
            DB2Command MyDB2Command = null;
            // Use the dsn alias that you defined in db2dsdriver.cfg with the db2cli writecfg command in step 1.
            String MyDb2ConnectionString = "database=alias;uid=userid;pwd=password;";
```

```

DB2Connection MyDb2Connection = new DB2Connection(MyDb2ConnectionString);
MyDb2Connection.Open();
MyDB2Command = MyDb2Connection.CreateCommand();
MyDB2Command.CommandText = "SELECT branch_code, city from GOSALES.BRANCH";
Console.WriteLine(MyDB2Command.CommandText);

DB2DataReader MyDb2DataReader = null;
MyDb2DataReader = MyDB2Command.ExecuteReader();
Console.WriteLine("BRANCH\tCITY");
Console.WriteLine("=====");
while (MyDb2DataReader.Read())
{
    for (int i = 0; i <= 1; i++)
    {
        try
        {
            if (MyDb2DataReader.IsDBNull(i))
            {
                Console.Write("NULL");
            }
            else
            {
                Console.Write(MyDb2DataReader.GetString(i));
            }
        }
        catch (Exception e)
        {
            Console.Write(e.ToString());
        }
        Console.Write("\t");
    }
    Console.WriteLine("");
}
MyDb2DataReader.Close();
MyDB2Command.Dispose();
MyDb2Connection.Close();
}
}
}

```

ODBC or CLI

Define a connection between a Microsoft Windows ODBC or CLI application and a IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. In a command shell on Linux operating systems, at the Windows command prompt, or in the Db2 command window on Windows operating systems, enter the following commands:

Note: These commands create new entries in the driver configuration file (`db2dsdriver.cfg`) on your computer and set the connection attributes. You need to do this step only one time.

- For a connection with SSL:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50001 -parameter "SecurityTransportMode=SSL"
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50001
```

- For a connection without SSL:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50000
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50000
```

where:

`<hostname>` is the host name of the server

`<alias>` is a DSN alias that you choose

2. (Optional): To test the connection to the database, run this command from the command prompt:

```
db2cli validate -dsn <alias> -connect -user <user_id> -passwd <password>
```

where:

`<alias>` is the DSN alias that you created with the `db2cli writecfg` command

`<user_id>` is from the connect credentials that you collected beforehand

`<password>` is from the connect credentials that you collected beforehand

3. (Optional): To register the data source name (DSN) with Microsoft ODBC Driver Manager and to work with Microsoft ODBC applications, run the following command. By default, the DSN is created as user DSN.

```
db2cli registerdsn -add -dsn <alias>
```

where:

`<alias>` is the DSN alias that you created with the `db2cli writecfg` command

ODBC Data Source Administrator

Use the Microsoft ODBC Data Source Administrator tool to define a connection between an ODBC or CLI application and a IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. Install the [Db2 driver package](#).
2. Open ODBC Data Source Administrator and create either a User DSN or System DSN for the Db2 driver package.
3. Click **Advanced** Settings. Enter following CLI parameters with their values for the IBM Db2 Warehouse SaaS server: **Hostname**, **Port**, and **Database**.
4. For an SSL connection, enter CLI parameter **Security** with value `SSL`.
5. Click **Apply**.
6. To test the connection, return to main page of ODBC Data Source Administrator. Click **Configure** for the DSN that you created. Enter the user ID and password for the server and click **Connect**.

PHP

Define a connection between a PHP application and a IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

Scenario 1: Connecting from outside IBM Cloud:

1. Download the [Db2 driver package](#) from the web console, and then install the driver package on the machine where your PHP application will run.
2. Use the [odbc_connect function](#) to connect to the BLUDB database.

PHP code example:

```
<?php
$database = "BLUDB";          # Get these database details from
$hostname = "<Host-name>";    # the web console
$user      = "<User-ID >";    #
$password  = "<Password>";    #
$port     = 50000;           #
```

```

$ssl_port = 50001;      #
# Build the connection string
#
$driver = "DRIVER={IBM DB2 ODBC DRIVER}";
$dsn    = "DATABASE=$database; " .
        "HOSTNAME=$hostname; " .
        "PORT=$port; " .
        "PROTOCOL=TCPIP; " .
        "UID=$user; " .
        "PWD=$password;";
$ssl_dsn = "DATABASE=$database; " .
        "HOSTNAME=$hostname; " .
        "PORT=$ssl_port; " .
        "PROTOCOL=TCPIP; " .
        "UID=$user; " .
        "PWD=$password; " .
        "SECURITY=SSL;";
$conn_string = $driver . $dsn;      # Non-SSL
$conn_string = $driver . $ssl_dsn; # SSL
# Connect
#
$conn = odbc_connect( $conn_string, "", "" );
if( $conn )
{
    echo "Connection succeeded.";
    # Disconnect
    #
    odbc_close( $conn );
}
else
{
    echo "Connection failed.";
}
?>

```

Saving this PHP code example to a script file called `C:\sample.php` and then running the script from a command line produces the following output:

```

$ C:\> php -f sample.php

Connection succeeded.

```

Scenario 2: Connecting from a PHP web app in IBM Cloud

1. From the IBM Cloud catalog, create a new PHP App.
2. From the "Getting Started" section of the new PHP App in your IBM Cloud dashboard, download the App starter code to your local work directory.
3. In your IBM Cloud dashboard, create a new connection from your Db2 service to your new PHP App. (Creating this Connection in IBM Cloud makes the `VCAP_SERVICES` environment variable available to your PHP App. The `VCAP_SERVICES` environment variable contains database details for your Db2 service. Using `VCAP_SERVICES` is more convenient than hardcoding the database details in your PHP App.)
4. In your local working directory, update the `index.php` file to connect to the BLUDB database by using the [db2_connect function](#).

Example:

```

<!DOCTYPE html>
<html>
<head>
    <title>PHP Starter Application</title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <link rel="stylesheet" href="style.css" />
</head>
<body>
<?php
if( getenv( "VCAP_SERVICES" ) )
{
    # Get database details from the VCAP_SERVICES environment variable
    #
    # *This can only work if you have used the Bluemix dashboard to

```



```

# create a connection from your dashDB service to your PHP App.
#
$details = json_decode( getenv( "VCAP_SERVICES" ), true );
$dsn      = $details [ "dashDB" ][0][ "credentials" ][ "dsn" ];
$ssl_dsn  = $details [ "dashDB" ][0][ "credentials" ][ "ssldsn" ];
# Build the connection string
#
$driver = "DRIVER={IBM DB2 ODBC DRIVER}";
$conn_string = $driver . $dsn;      # Non-SSL
$conn_string = $driver . $ssl_dsn; # SSL
$conn = db2_connect( $conn_string, "", "" );
if( $conn )
{
    echo "<p>Connection succeeded.</p>";
    db2_close( $conn );
}
else
{
    echo "<p>Connection failed.</p>";
}
}
else
{
    echo "<p>No credentials.</p>";
}
?>
</body>
</html>

```

5. From your local working directory, push the updates to IBM Cloud by following the instructions in the "Getting Started" section of the PHP App in your IBM Cloud dashboard. Then, restart the App in IBM Cloud and view the App in a browser.

Connecting to IBM tools

You can connect IBM command-line interfaces, applications, and tools to your IBM Db2 Warehouse SaaS database.

Data integration

InfoSphere DataStage

These instructions explain how to define a connection without SSL between IBM® InfoSphere® DataStage® and a IBM Db2 Warehouse SaaS database by cataloging the database and defining a connection object, or how to create a connection with SSL by using a digital certificate that is issued by a third party.

Prerequisites



Important: It is advised that you update DataStage to the most recent version so that you can take advantage of external tables to load your data into IBM Db2 Warehouse SaaS.

If you don't already have a data server client that is installed, download and install the IBM Data Server Client that is appropriate for your client machine's operating system: [IBM Data Server Client](#).

To make connections with the SSL protocol, download and install the 32-bit GSKit V8. Click the OS tab that is appropriate for your client machine's operating system: [GSKit V8 - Install, Uninstall and Upgrade instructions](#). For the following operating systems, ensure that you add the GSKit installation directory path to the OS-specific path environment variable:

- AIX®: **LIBPATH**
 - `/usr/opt/ibm/gsk8/lib`
- Linux: **LD_LIBRARY_PATH**
 - `/usr/local/ibm/gsk8/lib`
- UNIX: **LD_LIBRARY_PATH**
 - `/opt/ibm/gsk8/lib`
- Windows: **PATH**
 - `<installation_directory>\gsk8\bin`
 - `<installation_directory>\gsk8\lib`

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

- To create a connection with SSL, complete the following steps:

1. Open a command line or terminal and make a new directory in the DataStage system to store the SSL certificate and key files.

```
# /home/db2inst2> mkdir SSL
```

```
# /home/db2inst2> cd SSL
```

2. In the IBM Db2 Warehouse SaaS web console, download the SSL certificate from the **Connect your applications to the database** page.

a. From the main menu, click **Connect**.

b. Click **Connection with SSL**, and then click the **SSL certificate (1 KB)** link.

c. Save the `DigiCertGlobalRootCA.crt` certificate into the SSL directory that you made in step 1.

3. Create a client keystore database in the DataStage system by using the `gsk8capicmd_64` utility.

```
# /home/db2inst2/SSL> gsk8capicmd_64 -keydb -create -db <keystore_db.kdb> -pw <ks_db_password> -stash
```

where `<keystore_db.kdb>` represents the client keystore database and `<ks_db_password>` represents the password for the client keystore database.

4. Add the certificate to the client keystore database.

```
# /home/db2inst2/SSL> gsk8capicmd_64 -cert -add -db <keystore_db.kdb> -pw <ks_db_password> -label BLUDB_SSL -file DigiCertGlobalRootCA.crt
```

where `<keystore_db.kdb>` represents the client keystore database and `<ks_db_password>` represents the password for the client keystore database.

5. Configure the Db2 client on the DataStage server.

a. Update the SSL configuration parameters in the database manager.

```
# /home/db2inst2> db2 update dbm cfg using SSL_CLNT_KEYDB /home/db2inst2/SSL/<keystore_db.kdb>
```

where `<keystore_db.kdb>` represents the client keystore database.

```
# /home/db2inst2> db2 update dbm cfg using SSL_CLNT_STASH /home/db2inst2/SSL/<keystore_db.sth>
```

where `<keystore_db.sth>` represents the client keystore database password stash.

b. Catalog the target node with the SSL security option and then the BLUDB database at that target node.

```
# /home/db2inst2> db2 catalog tcpip node <node_name> remote <IP_addr_of_BLUDB_database_server> server 50001 security SSL
```

where `<node_name>` represents your name for the target node and `<IP_addr_of_BLUDB_database_server>` represents the IP address of the BLUDB database server,

```
# /home/db2inst2> db2 catalog db BLUDB as <db_alias> at node <node_name>
```

where `<db_alias>` is your name for the IBM Db2 Warehouse SaaS database.

6. Add read and execute permissions on the files in the SSL directory for everyone. The DataStage user who runs the jobs needs to access these files to make SSL connections to the Db2 database.

```
# /home/db2inst2/SSL> chmod 655 /home/db2inst2/SSL/*
```

7. Test the SSL connection in one of the following ways:

- Test the connection by using CLP. Issue the following command to connect to the IBM Db2 Warehouse SaaS database:

```
db2 connect to <db_alias> user <user_id>
```

where `<db_alias>` is your name for the IBM Db2 Warehouse SaaS database and `<user_id>` is your IBM Db2 Warehouse SaaS user ID. You are prompted to enter your password.

- Test the connection by using CLI. Issue the following command to connect to the IBM Db2 Warehouse SaaS database:

```
db2cli validate -dsn <alias> -connect -user <user_id> -passwd <password>
```

where `<alias>` is an alias that you created by using the `db2cli writecfg` command, `<user_id>` is your IBM Db2 Warehouse SaaS user ID, and `<password>` is your IBM Db2 Warehouse SaaS password.

- To create a connection without SSL, catalog the target IBM Db2 Warehouse SaaS database by completing the following steps:

1. Catalog the target IBM Db2 Warehouse SaaS node so that client applications can connect to it. Run the following CLP commands:

```
db2 catalog tcpip node <node_name> remote <IP_address_of_BLUDB_database_server> server <port_number_of_BLUDB_database>
```

where `<node_name>` represents your name for the node, `<IP_address_of_BLUDB_database_server>` represents the IP address of the BLUDB database server, and `<port_number_of_BLUDB_database>` represents the port number of the BLUDB database.

2. Catalog the remote IBM Db2 Warehouse SaaS database so that client applications can connect to it. Run the following command:

```
db2 catalog db BLUDB as <db_alias> at node <node_name>
```

where `<db_alias>` represents your name for the IBM Db2 Warehouse SaaS database and `<node_name>` represents your name for the node.

3. Test the non-SSL connection in one of the following ways:

```
$ - Test the connection by using CLP. Issue the following command to connect to the IBM Db2 Warehouse SaaS database:
```

```
`db2 connect to <db_alias> user <user_id>`
```

where `<db_alias>` is your name for the IBM Db2 Warehouse SaaS database and `<user_id>` is your IBM Db2 Warehouse SaaS user ID. You are prompted to enter your password.

```
`db2 list tables`
```

```
- Test the connection by using CLI. Issue the following command to connect to the IBM Db2 Warehouse SaaS database:
```

```
`db2cli validate -dsn <alias> -connect -user <user_id> -passwd <password>`
```

where `<alias>` is an alias that you created by using the `**db2cli writecfg**` command, `<user_id>` is your IBM Db2 Warehouse SaaS user ID, and `<password>` is your IBM Db2 Warehouse SaaS password.

4. Use the [connection information](#) that you collected beforehand to define a connection in the DataStage client. On the **Parameters** tab, you must select the **DB2 Connector** for the **Connect using Staging Type** field.

For details about defining a connection in DataStage, see the following DataStage documentation topics:

- [Creating a data connection object manually](#)
- [Configuring access to Db2 databases](#)

Lift

Use Lift to migrate your data into IBM Db2 Warehouse SaaS.

[Lift](#)

InfoSphere Data Replication

You can connect IBM® InfoSphere® Data Replication to a IBM Db2 Warehouse SaaS database. This capability applies to both SMP and MPP environments.

Overview

Ideally, when you connect IBM InfoSphere Data Replication to IBM Db2 Warehouse SaaS, IBM InfoSphere Data Replication is in the same IBM Cloud Data Center as IBM Db2 Warehouse SaaS or is colocated with IBM Db2 Warehouse SaaS. IBM InfoSphere Data Replication connects from a local server to the remote IBM Db2 Warehouse SaaS instance.

When you use IBM Db2 Warehouse SaaS as a connection target, the performance of IBM InfoSphere Data Replication partly depends on the bandwidth of the network that separates its target engine from the IBM Db2 Warehouse SaaS instance. Physical distance also affects performance: ideally, IBM InfoSphere Data Replication is as close as possible to the IBM Db2 Warehouse SaaS instance. Network topology also affects performance. For example, ideally, the IBM InfoSphere Data Replication target engine runs on a VM in the same VPN (security domain) as the target instance. The fewer the network nodes (for example, firewalls or routers) to traverse, the better.

Prerequisites

If you intend to connect by using the SSL protocol, download and install GSKit V8. See [GSKit V8 - Install, Uninstall and Upgrade instructions](#). Click the operating system tab that applies to your client machine's operating system. If you are installing the GSKit on a Windows computer, ensure that you specify

the GSKit installation directory path (`<installation_directory>\gsk8\bin`) for the `PATH` environment variable.

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

If you intend to connect by using the SSL protocol, download the `DigiCertGlobalRootCA.crt` SSL certificate from the web console to a directory on the client machine. To download the certificate, click **Connection > Connection Information** and then click the **Connection with SSL** tab.

Procedure

1. Choose one of the following approaches to make your connection:

- To create a connection with SSL, complete the following steps:

a. Issue the following command:

```
cd /<ssl_directory_name>/ssl
```

where `/<ssl_directory_name>/ssl` is the path to the directory into which you downloaded the `DigiCertGlobalRootCA.crt` SSL certificate.

b. Create a client key database and a stash file by using the `GSKCapiCmd` tool. For example, the following command creates a client key database that is called `dashclient.kdb` and a stash file that is called `dashclient.sth`:

```
gsk8capicmd_64 -keydb -create -db "dashclient.kdb" -pw "passw0rdpw0" -stash
```

where `passw0rdpw0` is a password. The `-stash` option creates a stash file in the same path as that of the client key database, with a file extension of `.sth`. At connection time, GSKit uses the stash file to obtain the password to the client key database.

c. Add the certificate to the client key database. For example, the following `gsk8capicmd` command imports the certificate from the `/<ssl_directory_name>/ssl/DigiCertGlobalRootCA.crt` file into the client key database called `dashclient.kdb`:

```
gsk8capicmd_64 -cert -add -db "dashclient.kdb" -pw "passw0rdpw0" -label "DigiCert" -file  
"/<ssl_directory_name>/ssl/DigiCertGlobalRootCA.crt" -format ascii -fips
```

d. Update the values of the `SSL_CLNT_KEYDB` and `SSL_CLNT_STASH` database manager configuration parameters on the client to specify the client key database and the stash file. Examples follow:

```
db2 update dbm cfg using SSL_CLNT_KEYDB /<ssl_directory_name>/ssl/dashclient.kdb
```

```
db2 update dbm cfg using SSL_CLNT_STASH /<ssl_directory_name>/ssl/dashclient.sth
```

e. Catalog the IBM Db2 Warehouse SaaS node so that client applications can connect to it. Issue the following command:

```
db2 catalog tcpip node <node_name> remote <Db2_Warehouse_IP_address> server <port_number> security ssl
```

where:

`<node_name>` is your name for the node.

`<Db2_Warehouse_IP_address>` is the IP address of the Db2 Warehouse on Cloud server.

`<port_number>` is the port that is used to connect to Db2 Warehouse by using an SSL connection. If you are using the default port, specify `50001`.

f. Catalog the remote IBM Db2 Warehouse SaaS database so that client applications can connect to it. Issue the following command:

```
db2 catalog database bludb as <db_alias> at node <node_name>
```

where `db_alias` is your name for the IBM Db2 Warehouse SaaS database.

g. Test the SSL connection in one of the following ways:

- Test the connection by using CLP by issuing the following command to connect to the IBM Db2 Warehouse SaaS database:

```
db2 connect to <db_alias> user <user_id>
```

where `<user_id>` is your IBM Db2 Warehouse SaaS user ID. You are prompted to enter your password.

- Test the connection by using CLI by issuing the following command to connect to the IBM Db2 Warehouse SaaS database:

```
db2cli validate -dsn <alias> -connect -user <user_id> -passwd <password>
```

where `<alias>` is a DSN alias that you created by using the `db2cli writecfg` command, `<user_id>` is your IBM Db2 Warehouse SaaS user ID, and `<password>` is your IBM Db2 Warehouse SaaS database password.

- To create a connection without SSL, complete the following steps:

a. Catalog the IBM Db2 Warehouse SaaS node so that client applications can connect to it. Issue the following command:

```
db2 catalog tcpip node <node_name> remote <Db2_Warehouse_IP_address> server <port_number>
```

where:

<node_name> is your name for the node.

<Db2_Warehouse_IP_address> is the IP address of the IBM Db2 Warehouse SaaS server.

<port_number> is the port that is used to connect to IBM Db2 Warehouse SaaS without using an SSL connection. If you are using the default port, specify **50000**.

b. Catalog the remote IBM Db2 Warehouse SaaS database so that client applications can connect to it. Issue the following command:

```
db2 catalog database bludb as <db_alias> at node <node_name>
```

where <db_alias> is your name for the IBM Db2 Warehouse SaaS database.

c. Test the non-SSL connection in one of the following ways:

- Test the connection by using CLP by issuing the following command to connect to the IBM Db2 Warehouse SaaS database:

```
db2 connect to <db_alias> user <user_id>
```

where <user_id> is your IBM Db2 Warehouse SaaS user ID. You are prompted to enter your password.

- Test the connection by using CLI by issuing the following command to connect to the IBM Db2 Warehouse SaaS database:

```
db2cli validate -dsn <alias> -connect -user <user_id> -passwd <password>
```

where <alias> is a DSN alias that you created by using the **db2cli writecfg** command, <user_id> is your IBM Db2 Warehouse SaaS user ID, and <password> is your Db2 Warehouse on Cloud password.

2. Launch the InfoSphere Data Replication configuration tool and perform the following steps. The values that are shown in the screen captures are examples.

a. Add a source instance to point to your source database by using the **Instance Configuration** tab:

IBM InfoSphere Data Replication New Instance

Instance Configuration | Communications Protocol

Instance

Name: source

Server Port: 10901

Auto_Discovery Port:

Staging Store Disk Quota (GB): 1

Maximum Memory Allowed (MB): 1024

Windows Service

Log on as:

Local System Account

This account: _____

Password: _____

Confirm Password: _____

Database

DB2 Instance: DB2

Name: sourcedb Advanced...

Username: db2user

Password: ●●●●●●

Metadata Schema: CDC_schema ...

Refresh Loader

Specify refresh loader path

Refresh Loader Path: _____ Browse...

OK Cancel

Figure 1. IIDR New Instance - Source instance

b. Add a target instance to point to your target Db2 database by using the **Instance Configuration** tab. If you are not using IBM InfoSphere Data Replication 11.3.3.3-50 or later, do not select the **Specify refresh loader path** check box.

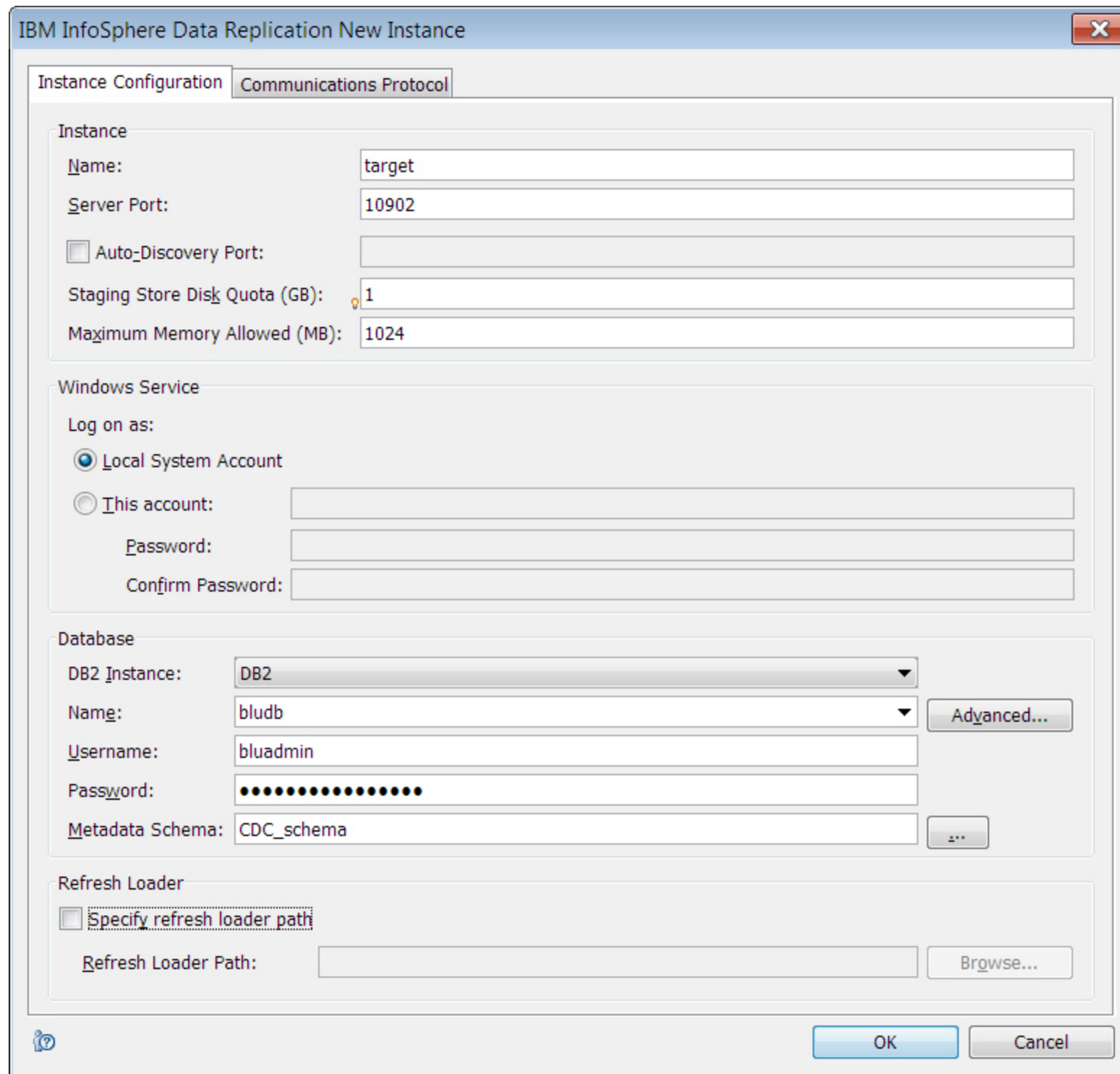


Figure 2. IIDR New Instance - Target instance

c. Start each instance:

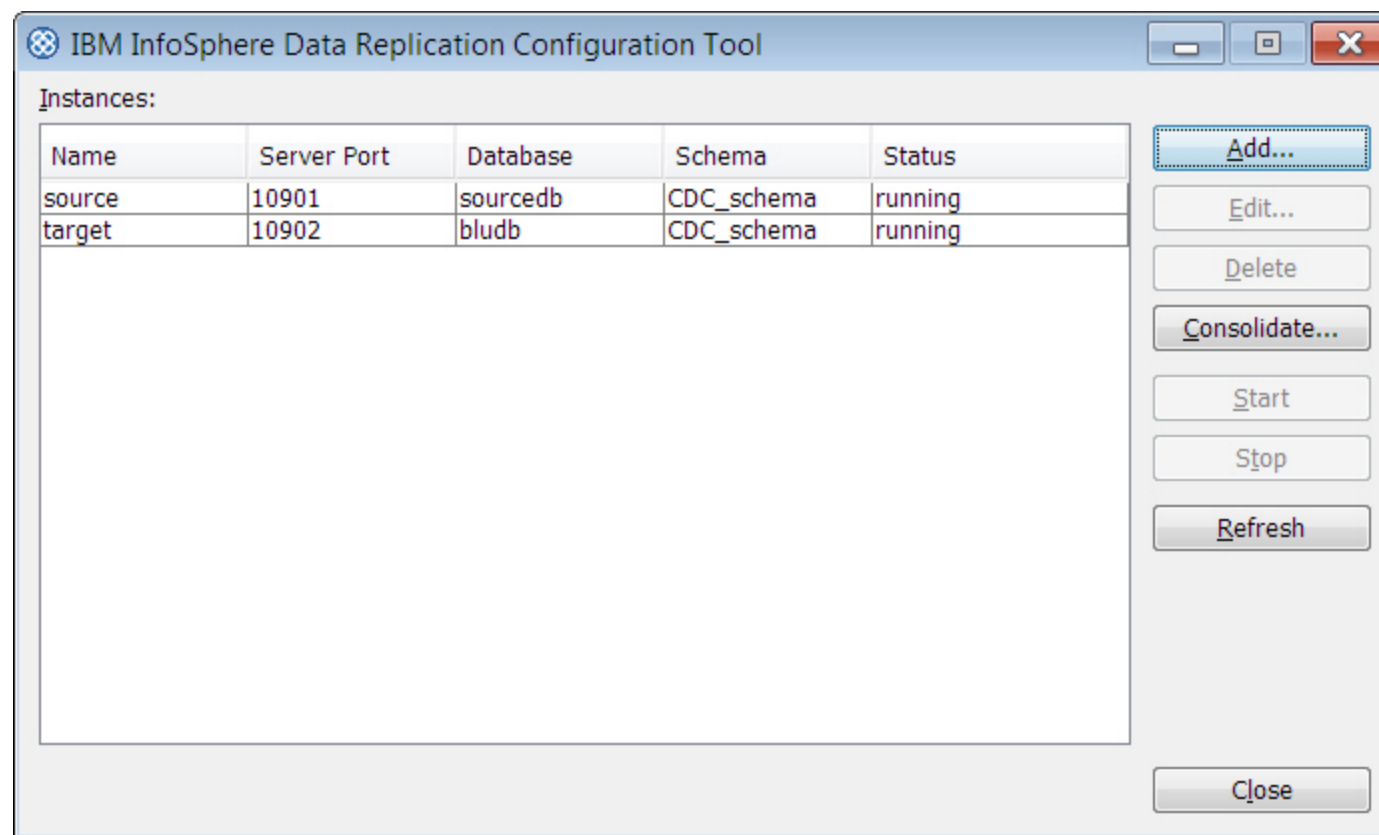


Figure 3. IIDR Configuration Tool

3. Launch the InfoSphere Data Replication management console and use Access Manager to complete the following steps:

a. Create a datastore to connect to your source instance by using the **Datastore** tab. Because a Db2 database was not originally supported as a source database, you must provide user and password information for the source database by clicking **Connection Parameters**.

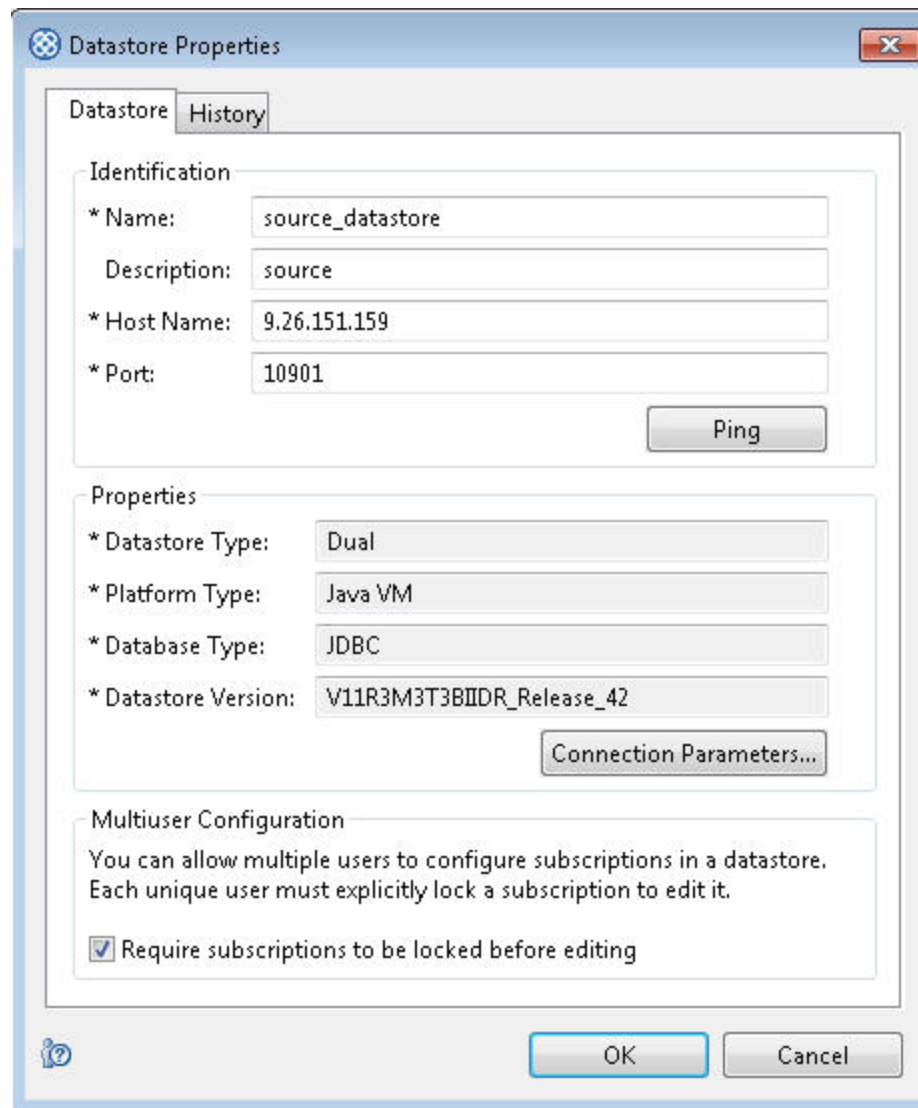


Figure 4. View of source datastore properties

b. Create a datastore to connect to your target instance by using the **Datastore** tab. You must provide user and password information by clicking **Connection Parameters**.

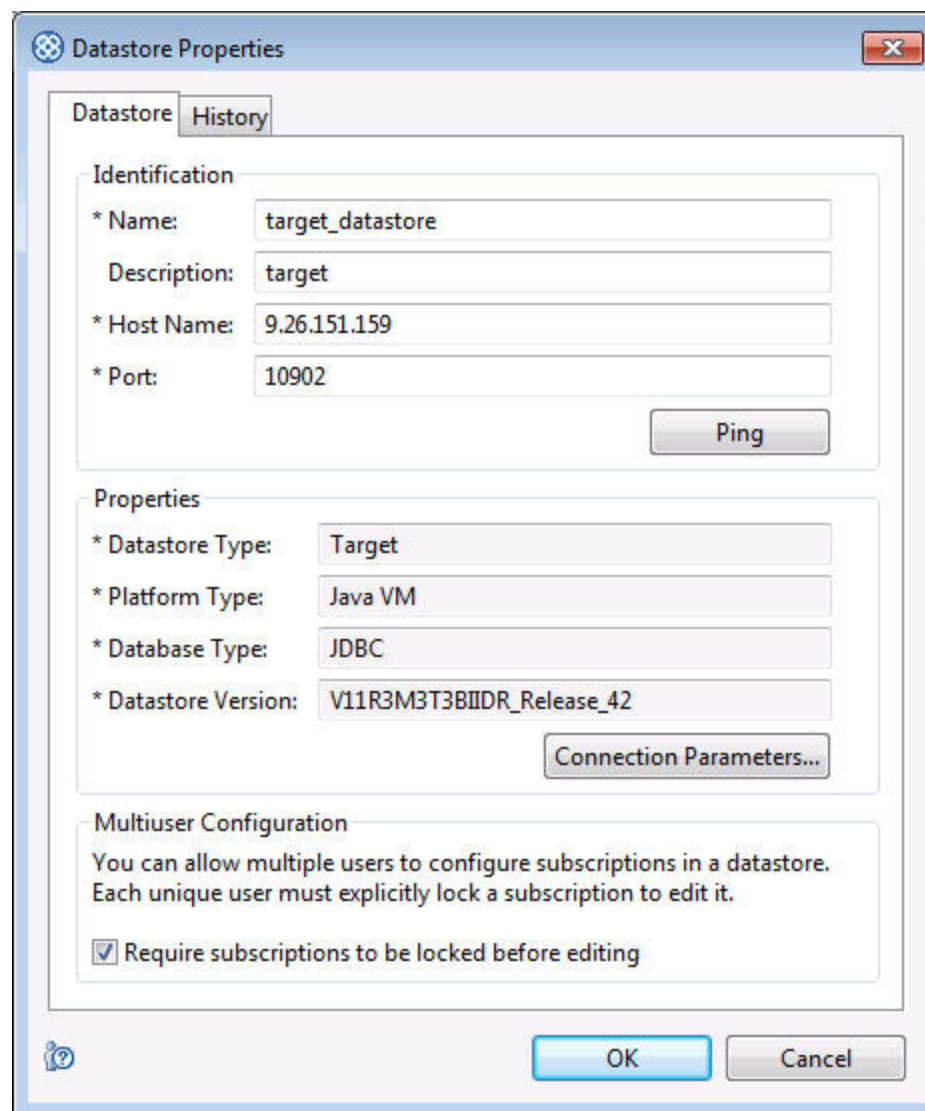


Figure 5. View of target datastore properties

c. If the user (for example, admin) that connects to the Access Server does not exist, create that user:

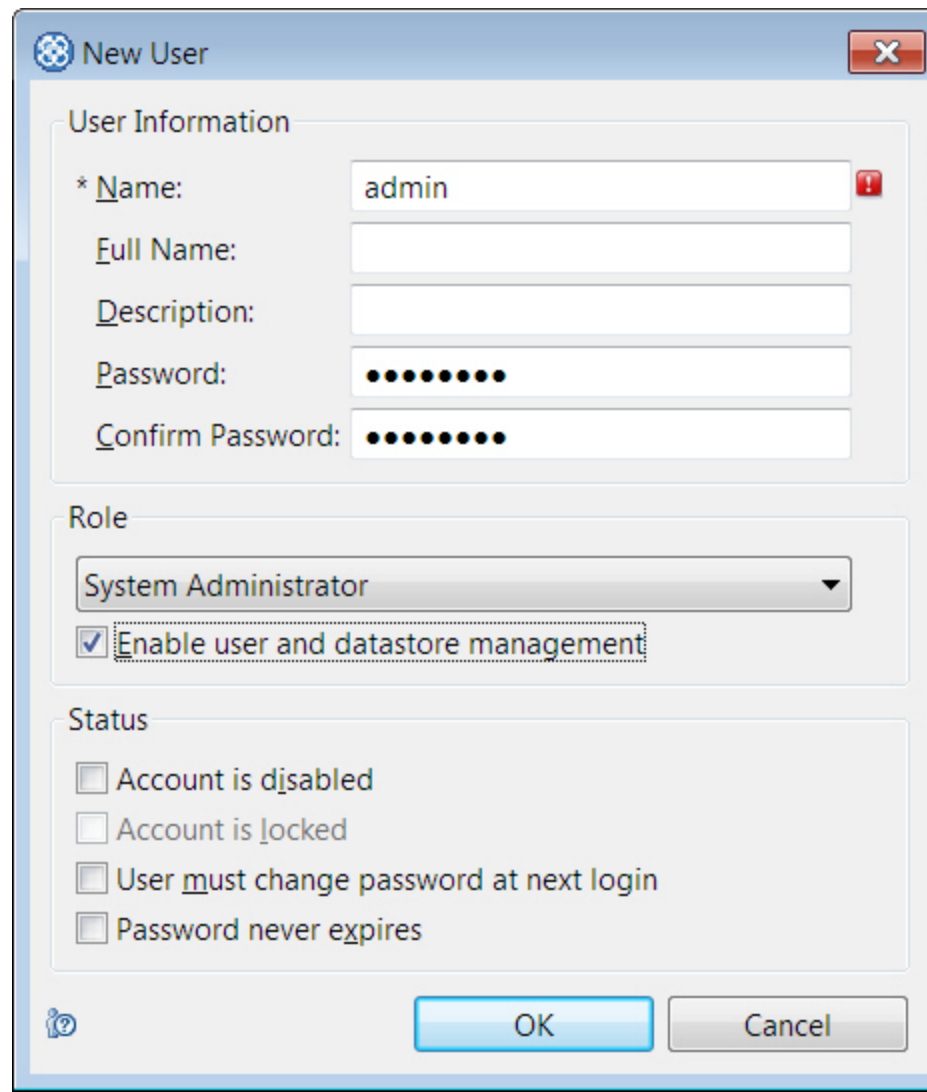


Figure 6. View of New User creation tool

d. Click the **Access Manager** tab.

e. On the **Datastore Management** tab, assign the user to both the source and target datastores by right-clicking each datastore and then clicking **Assign User**. Ensure that the credentials for accessing each instance are correct.

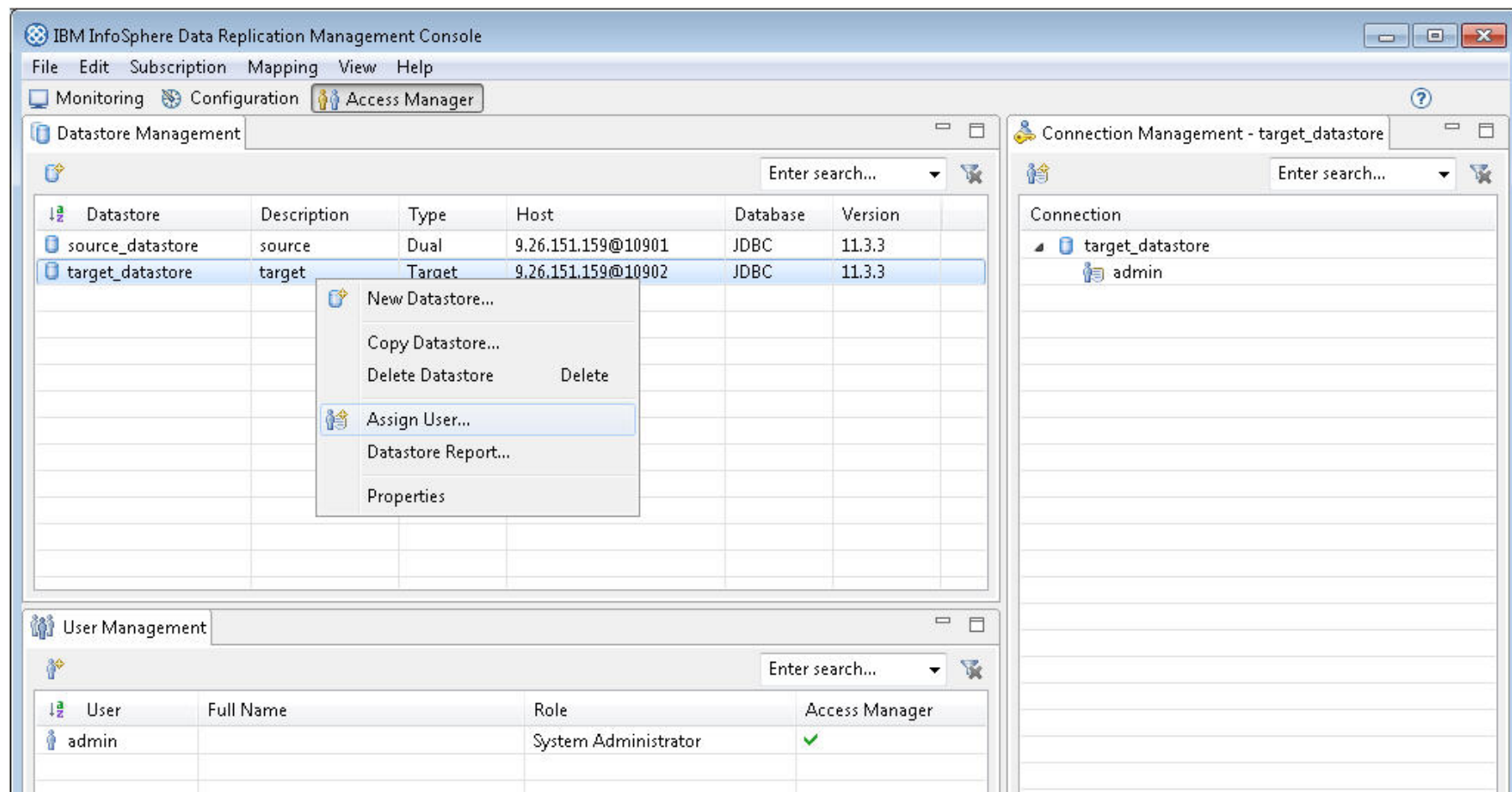


Figure 7. IIDR Management Console - Access Manager

What to do next

Define a subscription and perform data replication. For information, see:

- [Loading data from InfoSphere Data Replication](#)

Data Studio

These instructions explain how to create a connection from IBM® Data Studio to a IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. In Data Studio, click **All Databases > New Connection to a database**.
2. On the **Local** tab, select **DB2 for Linux, UNIX, and Windows** as the database manager.
3. On the **General** tab, enter the following values:
 - *Database*: **BLUDB**
 - *Host*: The host name.
 - *Port*: For a connection without SSL, enter **50000**. For a connection with SSL, enter **50001**.
 - *User name*: The user name that you use to log in.
 - *Password*: The password that you use to log in.
4. For an SSL connection, click the **Optional** tab, and then click **Add**. For the **sslConnection** property, specify **true**.
5. (Optional): Click **Test Connection** to verify that the connection succeeded.

Data Server Manager (DSM)


A connection between your IBM® Data Server Manager and your IBM Db2 Warehouse SaaS database enables you to monitor and manage the database from the Data Server Manager web console.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

To create a connection, complete the following steps:

1. Log in to your Data Server Manager web console.
2. In the Data Server Manager web console, go to **Set Up > Database Connections**.
3. Click the  icon to add a database connection. On the **Add Database Connection** page under the **Database Connection** tab, enter the required information in the following fields:
 - *Database connection name*: The name must be unique to Data Server Manager
 - *Data server type*: From the drop-down menu, select **DB2 for Linux, UNIX, and Windows**
 - *Database name*: **BLUDB**
 - *Host name*: Enter the IBM Db2 Warehouse SaaS host name
 - *Port number*: For a connection without SSL, enter **50000**. For a connection with SSL, enter **50001**.
 - *JDBC security*: From the drop-down menu, select **Clear text password**
 - *User ID*: Your IBM Db2 Warehouse SaaS user ID
 - *Password*: Your IBM Db2 Warehouse SaaS password
4. For a connection with SSL, select the **Advanced JDBC Properties** tab. Enter the required information in the following fields:
 - *Property*: **sslConnection**
 - *Value*: **true**

Click the **Add** button. Select the **Database Connection** tab.

5. Test the connection by clicking the **Test Connection** button. If the connection is successful, click **OK**.

InfoSphere Data Architect

These instructions explain how to create a connection from InfoSphere® Data Architect to a IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. In the Data Source Explorer view of InfoSphere Data Architect, right-click **Database Connections**, then select **New**.

2. On the **Local** tab, select **DB2 for Linux, UNIX, and Windows** as the database manager.
3. On the **General** tab, enter the following values:
 - *Database*: **BLUDB**
 - *Host*: The host name that you collected beforehand.
 - *Port*: For a connection without SSL, enter **50000**. For a connection with SSL, enter **50001**.
 - *User name*: The user ID that you collected beforehand.
 - *Password*: The password that you collected beforehand.
4. For an SSL connection, click the **Optional** tab. Enter an **sslConnection** property and specify a value of **true**. Click **Add**.
5. (Optional): Click **Test Connection** to verify that the connection succeeded.

CLPPlus

Command line processor plus (CLPPlus) is included in the Db2 driver package. CLPPlus provides a command-line interface that you can use to connect to a IBM Db2 Warehouse SaaS database. You can use CLPPlus to define, edit, and run statements, scripts, and commands.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

To use CLPPlus, ensure that a software development kit (SDK) or a Java runtime environment (JRE) for Java Version 1.5.0 or later is installed on your computer and that environment variables are set as follows:

- The **JAVA_HOME** environment variable is set to the Java installation directory on your computer.
- The **PATH** environment variable setting includes the **bin** subdirectory of the Java installation directory on your computer.

Procedure

1. In a command shell on Linux operating systems, at the Windows command prompt, or in the DB2 command window on Windows operating systems, run the following commands:

These commands create new entries in the driver configuration file (**db2dsdriver.cfg**) on your computer and set the connection attributes. You need to do this step only one time.

- For a connection with SSL:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50001 -parameter "SecurityTransportMode=SSL"
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50001
```

where:

- **<hostname>** is the host name of the server.
- **<alias>** is an alias that you choose.

- For a connection without SSL:

```
db2cli writecfg add -database BLUDB -host <hostname> -port 50000
```

```
db2cli writecfg add -dsn <alias> -database BLUDB -host <hostname> -port 50000
```

2. To start CLPPlus with a connection to a IBM Db2 Warehouse SaaS database that uses the entries in the **db2dsdriver.cfg** file, run the following command:

- Windows environments:

```
clpplus <userid>@<alias>
```

- Linux environments:

```
clpplus -nw <userid>@<alias>
```

where:

- **<userid>** is the user ID from the connect credentials you collected beforehand.
- **<alias>** is the alias that you created with the **db2cli writecfg** command.

Running this command opens a CLPPlus window.

3. In the CLPPlus window, enter your password. The database information is displayed, followed by an SQL prompt. Sample output follows:

```
$ Hostname = 192.0.2.0
Database server = DB2/LINUX8664 SQL10054
SQL authorization ID = smith
Local database alias = BLUDB
Port = 50001

SQL>
```

Results

You can now enter CLPPlus commands or SELECT statements and run scripts to work with the data in the database.

Examples

The following examples use a short script that retrieves rows from the sample table `GOSALES.BRANCH`. The script file is named `cities.sql` and is on the local Windows computer in the `C:\temp directory`. The `cities.sql` file contains the following text:

```
$ SET ECHO ON
SELECT branch_code, city from GOSALES.BRANCH;
```

Example 1

To run the script interactively:

1. Start CLPPlus with your user ID and the alias that you created in the `db2dsdriver.cfg` file by running the following command:

```
clpplus <user_id>@<alias>
```

2. Enter your password.
3. At the SQL prompt, enter the following text:

```
start C:\temp\cities.sql
```

Example 2

Start CLPPlus with your user ID and the alias that you created in the `db2dsdriver.cfg` file and run the script in one step:

```
clpplus <user_id>/<password>@<alias> @C:\temp\cities.sql
```

Sample output from the `cities.sql` script follows:

```
$ BRANCH_CODE CITY
-----
      6 Paris
      7 Milano
      9 Amsterdam
     13 Hamburg
     14 München
     15 Kista
     17 Calgary
     18 Toronto
     19 Boston
     20 Seattle
     21 Los Angeles
     22 Miami
     23 Lyon
     24 Distrito Federal
     25 Tokyo
     26 Osaka City
     28 Melbourne
     29 Bilbao
     30 Sao Paulo
     31 Kuopio
     32 Seoul
     33 Singapore
```



```
BRANCH_CODE CITY
```

```
-----  
34 Shanghai  
35 London  
36 Birmingham  
37 Zürich  
38 Heverlee  
39 Wien  
40 Geneve
```

```
29 rows were retrieved.
```

Data visualization & BI

Cognos Analytics

You can run your IBM Cognos® reports against data in the cloud rather than the data in an on-premises database. After you load your data to a IBM Db2 Warehouse SaaS database, you set up the JDBC driver, and then use the Cognos administration tools to create the database connection.

Watch this video to see how to create a connection.



View video: [Creating a connection from Cognos Analytics](#)

-->

To create a new connection in a legacy project:

1. From your project's **Assets** page, click the **Find and add data** icon.
2. On the **Connections** pane, click **Create Connection**.
3. Enter a name and description and choose a Service Category:

Data Service = IBM Cloud data service

When you choose **Data Service**, your existing IBM Cloud data services appear in the **Service Instance** list.

4. Choose the service or database server from the list.

5. Enter the connection information:

When you choose **Data Service**, your existing IBM Cloud data services appear in the **Service Instance** list.

6. Click **Create**. The connection is available to all legacy projects.
7. On the **Connections** pane, select the connection and click **Apply**.

To add an existing connection to a legacy project:

1. From your project's **Assets** page, click the **Find and add data** icon.
2. On the **Connections** pane, select the connection and click **Apply**.

SPSS Statistics

These instructions explain how to create a connection from IBM® SPSS® Statistics to a IBM Db2 Warehouse SaaS database.

Prerequisites

Before attempting to connect to your IBM Db2 Warehouse SaaS database, verify that you have the [prerequisites](#).

Procedure

1. In SPSS Statistics, click **File > Open Database > New Query**.

2. In the Database Wizard, click **Add ODBC data source**.
3. Use the ODBC Data Source Administrator window to add an ODBC data source name (DSN) for the IBM Db2 Warehouse SaaS database:
 - a. On the **User DSN** tab, click **Add**.
 - b. On the Create New Data Source window, select the driver that is named **IBM DB2 ODBC DRIVER** and click **Finish**.

Be sure that the driver that you select is not a driver with a similar name such as **IBM DB2® ODBC DRIVER - DB2COPY**. If the driver does not appear in the list, exit SPSS Statistics, then download and install the driver. See [Driver package](#).
 - c. In the ODBC IBM Driver - Add window, enter a data source name (usually the name of the database you are connecting to), and click **Add**.
 - d. From the CLI/ODBC Settings window, on the **Data Source** tab, enter the user ID and password from the [connection information](#) that you collected beforehand.
 - e. On the Advanced Settings page, for each of the following parameters, click **Add** to open the Add CLI Parameter window to begin the step, and click **OK** to return to the Advanced Settings page:
 - Select the **Database** parameter, then click **OK** to open the CLI/ODBC Settings window. In the **Pending Value** field, enter the database name from the connection information that you collected beforehand.
 - Select the **Hostname** parameter, then click **OK** to open the CLI/ODBC Settings window. In the **Pending Value** field, enter the host name from the connection information that you collected beforehand.
 - Select the **Port** parameter, then click **OK** to open the CLI/ODBC Settings window. In the **Pending Value** field, enter the port number from the connection information that you collected beforehand.
 - Select the **Protocol** parameter, then click **OK** to open the CLI/ODBC Settings window. Select **TCP/IP**.
 - f. Click **OK** to return to the ODBC Data Source Administrator window.
 - g. On the **User DSN** tab, select the data source name, then click **Configure**.
 - h. On the CLI/ODBC Settings window, click **Connect** to test the connection.
 - If the connection is successful, click **OK** to return to the ODBC Data Source Administrator window, and then click **OK** to exit the window.
 - If the connection is not successful, note and correct any errors before you test the connection again.

Connecting with third-party tools

You can connect third-party command-line interfaces, applications, and tools to your IBM Db2 Warehouse SaaS database.

Data integration

Informatica

You can connect Informatica to IBM Db2 Warehouse SaaS to help you manage your data.

Watch this video to see how to integrate IBM Db2 Warehouse SaaS with Informatica Cloud.



View video: [DB2 Connections - Lightning Fast How-To with Informatica Cloud](#)

REST API

You can use the REST API to access data, view and create database objects, administer, scale storage and compute, and monitor your Db2 Warehouse on Cloud service.

- All plans hosted on IBM Cloud, and previous generation of plans hosted on AWS: [V4 REST API](#).
- Current generation of plans hosted on AWS: [V5 REST API](#).

Loading your data

Open data format table support

The current generation of IBM® Db2® Warehouse as a Service plans on AWS supports open data formats as DATALAKE tables, allowing for seamless access to other data within your enterprise for integrated workloads. You can:

- Browse, explore, and query enterprise data in both Db2 and DATALAKE formats, using either the web-based UI, or through SQL
- Access data in place within DATALAKE tables, joining as necessary with Db2 based data for queries
- Access data within DATALAKE tables and import into Db2 formatted tables
- Create new DATALAKE tables in S3 and export from Db2 formatted tables

Supported open table and data formats include Iceberg, Parquet, ORC, and CSV. Both regular and Iceberg DATALAKE table types are supported, based on existing data formats or for business/technical requirements such as ACID compliance. For more information about open data format support in IBM Db2 Warehouse SaaS, see [Using DATALAKE tables](#)

IBM® Db2® Warehouse as a Service also provides seamless integration and common metadata service with the new IBM watsonx.data service. You can use the web-based UI to register your watsonx.data service, to allow the import and export of data between the services. For more information about integration with watsonx.data, see [watsonx.data Integration](#)

Loading data from your local system

You can load data in a delimited format file (CSV or TXT) from your local system into IBM® Db2® Warehouse as a Service. For files less than 500 MB in size, simply drag and drop them onto the **Load** page in the IBM Db2 Warehouse SaaS web console.

When loading data through the web console, IBM Db2 Warehouse SaaS autodetects and infers column types for your file. You can also configure delimiters and date/time formats for your data.

If you want to load larger CSV files on the gigabyte or terabyte scale into IBM Db2 Warehouse SaaS on IBM Cloud, we recommend using [IBM Lift CLI](#), a free ground-to-cloud data migration tool.

Loading data from IBM Cloud Object Storage

External Tables

You can load data from IBM Cloud Object Storage (COS) into IBM® Db2® Warehouse as a Service by using the built-in External Tables functionality.

Here's an example SQL statement that inserts COS data into a IBM Db2 Warehouse SaaS table by using External Tables:

```
INSERT INTO <table-name> SELECT * FROM EXTERNAL '<mys3file.txt>' USING
(CCSID 1208 s3('s3-api.us-geo.objectstorage.softlayer.net',
 '<S3-access-key-ID>',
 '<S3-secret-access-key>',
 '<my_bucket>'
 )
 )
```



Note: For IBM Cloud Object Storage, to create HMAC credentials when creating new service credentials, specify {"HMAC:true"} in the *Add Inline Configuration Parameters* field.

Db2 LOAD utility

You can also load data from IBM Cloud Object Storage (COS) into IBM Db2 Warehouse SaaS by using the built-in Db2 **LOAD** utility.

Here's an example SQL statement that uses the Db2 LOAD utility to load COS data into IBM Db2 Warehouse SaaS:

```
CALL SYSPROC.ADMIN_CMD('LOAD FROM "S3::<cos-endpoint-url>::<cos-access-key-ID>::<cos-secret-access-key>:
:<cos-bucket-name>::<path-to-data-file>" OF <filetype> <additional-load-options> INTO <table-name>')
```

For more information about the Db2 LOAD utility, see: [LOAD command](#).

Loading data from Amazon S3

External Tables

You can load data from Amazon S3 into IBM® Db2® Warehouse as a Service by using the built-in External Tables functionality.

Here's an example SQL statement that inserts Amazon S3 data into a IBM Db2 Warehouse SaaS table by using External Tables:

```
INSERT INTO <table-name> SELECT * FROM EXTERNAL '<mys3file.txt>' USING
  (CCSID 1208 s3('s3.amazonaws.com',
  '<S3-access-key-ID>',
  '<S3-secret-access-key>',
  '<my_bucket>'
  )
  )
```

Db2 LOAD utility

You can also load data from Amazon S3 into IBM Db2 Warehouse SaaS by using the built-in Db2 **LOAD** utility.

Here's an example SQL statement that uses the Db2 LOAD utility to load Amazon S3 data into IBM Db2 Warehouse SaaS:

```
CALL SYSPROC.ADMIN_CMD('LOAD FROM "S3::<s3-endpoint-url>::<s3-access-key-ID>::<s3-secret-access-key>:
:<s3-bucket-name>::<path-to-data-file>" OF <filetype> <additional-load-options> INTO <table-name>')
```

For more information about the Db2 LOAD utility, see: [LOAD command](#).

Loading data from on-premises systems with IBM Lift CLI

[IBM Lift CLI](#) is a free ground-to-cloud data migration tool that securely moves your data from on-premises systems to IBM® Db2® Warehouse as a Service plans hosted on IBM Cloud. IBM Lift CLI is available for the Flex One, Flex, and Flex Performance plans hosted on IBM Cloud.

The Lift application migrates your data to the IBM Cloud from the various data sources listed in Table 1.


Target database on IBM Cloud	Data source
IBM Db2 Warehouse SaaS	IBM Db2
	IBM Db2 Warehouse
	IBM Integrated Analytics System
	IBM PureData System for Analytics
	Oracle Database
	Microsoft SQL Server
	CSV file format

Table 1. Migration data sources

When your data source is IBM Db2, IBM Db2 Warehouse, IBM Integrated Analytics System, or IBM PureData System for Analytics, migration of your data with Lift happens in two phases:

- Migrating the table structure of the source database
- Moving data into IBM Db2 Warehouse SaaS

Migrating Table Structure

 **Important:** The Lift application may accept obsolete TLS protocol in SSL handshake if the server it is connecting to does not support TLS version 1.2. To ensure TLS 1.0 and TLS 1.1 will not be in use during communication, edit `liftMain` file adding in lines 115 and 118 `-Djdk.tls.client.protocols=TLSv1.2 -Dhttps.protocols=TLSv1.2` to Java™ invocation as specified [here](#).


Migrate your table structure by using the following lift ddl command example, which extracts the Data Definition Language (DDL) from the source database

and applies it to the target database:

```
% lift ddl --migrate --source-schema <source-schema-name> --source-object <source-object-name>
--source-database <source-database-name> --source-user <source-user-name>
--source-password <source-password> --source-host <source-database-host-name>
--source-database-port <source-database-port> --source-database-type <source-database-type>
--target-user <target-user-name> --target-password <target-password>
--target-host <target-database-host-name>
```

For more available command options, run the lift ddl --help command.

Moving Data

 **Important:** The Lift application may accept obsolete TLS protocol in SSL handshake if the server it is connecting to does not support TLS version 1.2. To ensure TLS 1.0 and TLS 1.1 will not be in use during communication, edit `liftMain` file adding in lines 115 and 118 `Djdk.tls.client.protocols=TLSv1.2 -Dhttps.protocols=TLSv1.2` to Java™ invocation as specified [here](#).

Extract the table data into a CSV file by running the following lift extract command example:

```
% lift extract --source-schema <source-schema-name> --source-table <source-table-name>
--source-database <source-database-name> --source-host <source-host-name>
--source-user <source-user-name> --source-password <source-password>
--source-database-port <source-database-port> --source-database-type <ias/db2/db2w>
--file <path-to-csv-file>
```

The ias, db2, and db2w settings for the `--source-database-type` command option are used to specify the particular source database type.

- Move the CSV file and stage it in the landing zone of the target database by running the following lift put command example:

```
% lift put --file <path-to-csv-file> --target-user <target-user-name>
--target-password <target-password> --target-host <target-database-host-name>
```

- Load the data from the CSV file into the target database by running the following lift load command example:

```
% lift load --filename <csv-file-name> --target-schema <target-schema-name>
--target-table <target-table-name> --file-origin <extract-ias/extract-db2/extract-db2w>
--target-user <target-user-name> --target-password <target-password>
--target-host <target-database-host-name>
```

The extract-ias, extract-db2, and extract-db2w settings for the `--file-origin` command option are used to specify that the CSV file was extracted from a particular database by using the lift extract command.

After completing these steps, you can run queries against your data.

After your table structure is in place, you can start moving your data. Lift CLI extracts the data from your table into a CSV file, moves that file over the network and stages it in the landing zone of the target database, and then loads the data into the database.

To download and install Lift, see the IBM Db2 Download Center: [Download Lift](#).

Assessing your database compatibility

IBM Database Conversion Workbench (DCW) is a no-charge plug-in that adds database migration capabilities to IBM Data Studio. You can download both Data Studio and Database Conversion Workbench from the web console.

To assess the compatibility of your database with Db2 and convert the SQL, you can also use the free, light-weight tool called [IBM Database Harmony Profiler](#).

For more information about the Database Conversion Workbench and Database Harmony Profiler, see: [IBM Database Conversion Workbench \(DCW\)](#).

Distribution keys for multi-partitioned plans

When creating a table in one of the IBM® Db2® Warehouse as a Service [multi-partitioned plans](#), you can decide how you want to distribute data among data partitions.

You can use either [hash distribution](#) or [random distribution](#).

Hash distribution

With hash distribution, data is distributed by applying a hashing algorithm to the values in the columns that are listed in a *distribution key*. A well-chosen distribution key can balance two objectives:

- Maximizing parallel processing of queries and maximizing the use of available storage space by distributing table data evenly across the system, and
- Minimizing the time that it takes to fetch data by collocating data that is likely to be fetched together.

In general, a well-chosen hash distribution key produces the best performance. You can specify the distribution key by including the `DISTRIBUTE BY HASH` clause in the `CREATE TABLE` statement:

```
CREATE TABLE MYTABLE
(
  COL1 INT,
  COL2 VARCHAR(5)
)
DISTRIBUTE BY HASH( COL1 )
```

For more information about choosing hash distribution keys, see [Choosing a hash distribution key for a table in an MPP database](#).

Random distribution

With random distribution, data is distributed evenly across the system. Random distribution maximizes parallel processing of queries and maximizes the use of available storage space.

Consider using random distribution as a simpler alternative to hash distribution, for the following reasons:

1. If you don't have enough information to choose an effective hash distribution key,
2. If you know that collocation isn't needed for the queries that are entered against the table, or
3. If the table is small enough that performance is not affected by the distribution.

You can use random distribution by including the `DISTRIBUTE BY RANDOM` clause in the `CREATE TABLE` statement.

If you specify `DISTRIBUTE BY RANDOM` when creating a table with a primary or unique key, the database manager implements the random distribution by creating a hash key on the unique or primary key. In this case, when you view the table definition in the web console or the catalog tables, you'll see this hash distribution key despite the fact that you specified `DISTRIBUTE BY RANDOM` when you created the table.

```
CREATE TABLE MYTABLE
(
  COL1 INT,
  COL2 VARCHAR(5)
)
DISTRIBUTE BY RANDOM
```

Default behavior

If you do not specify a distribution clause when creating a table in a multi-partitioned plan, a default hash distribution key is used.

Administration

Interfaces

You can work with your warehouse database in the following ways:

- From the web console
- REST API
- Connect applications or your favorite tools from your local computer
- Use IBM Db2 Warehouse SaaS as a data source for your IBM Cloud apps or services

Web console

The web console provides a graphical interface for everything that you need to use your database, including: load facilities, an SQL editor, driver downloads, and more.

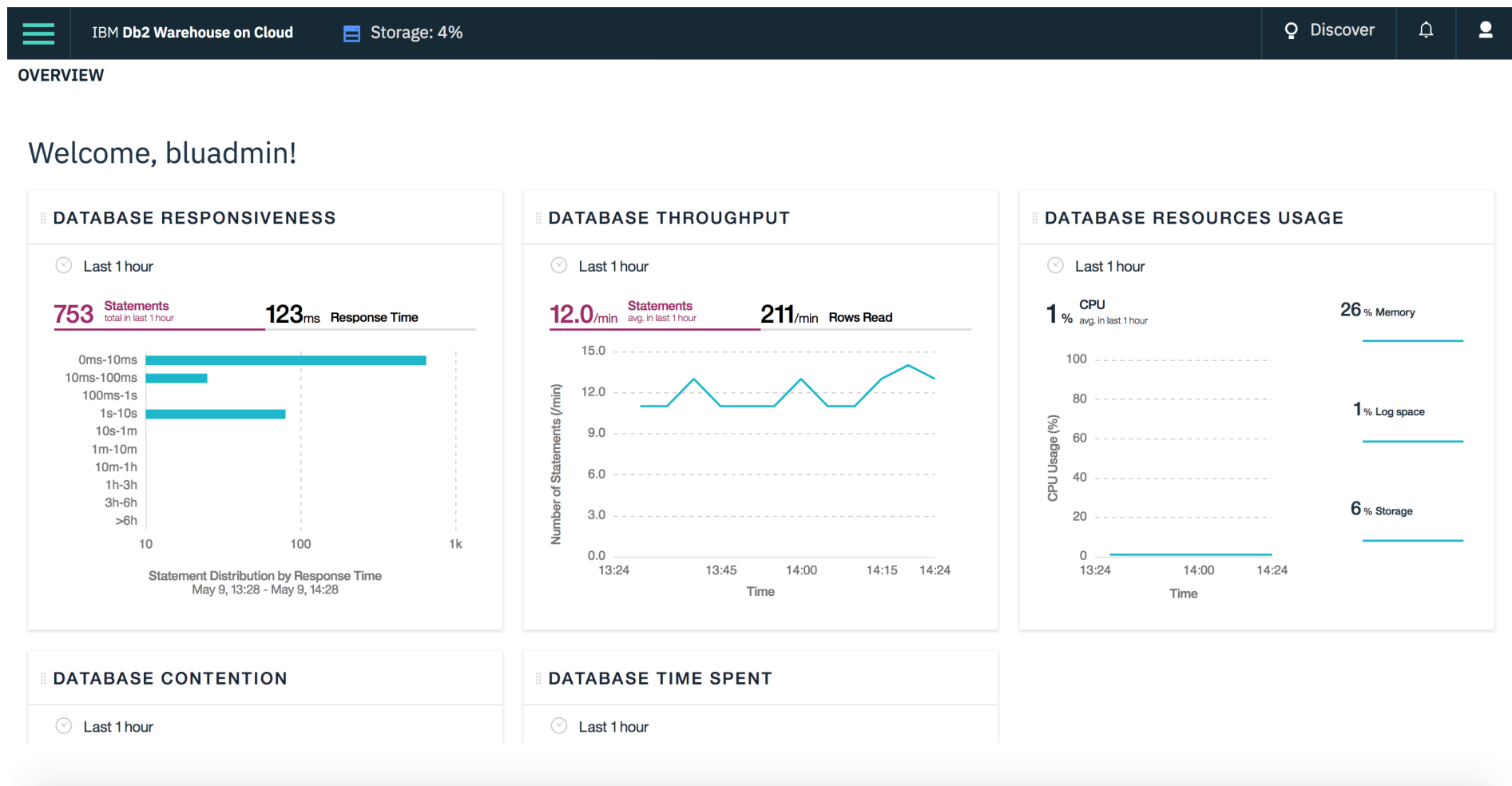


Figure 1. View of the web console dashboard page

You can access your web console in the following ways:

- From your IBM Cloud dashboard - You can open the web console from the Service Details page for your IBM Db2 Warehouse SaaS service.
- Direct URL - You can bookmark the URL of the web console for your IBM Db2 Warehouse SaaS service.

REST API

With Db2 Warehouse on Cloud service plans, you can do tasks that are related to file management, loading data, and resource scaling using the database management [REST API](#)

Connect applications or your favorite tools from your local computer

Configure your local environment to connect to your IBM Db2 Warehouse SaaS database by completing the following steps:

1. Download the [driver package](#) from the IBM Db2 Warehouse SaaS web console.
2. Install the driver package on the computer where your apps or tools are running:
 - [Installing on Linux or PowerLinux](#)
 - [Installing on Mac OS X](#)
 - [Installing on Windows](#)
3. [Configure the driver files](#) for your IBM Db2 Warehouse SaaS database.

Use Db2 Warehouse on Cloud as a data source for your IBM Cloud apps or services

Apps that are hosted on IBM Cloud can connect to your IBM Db2 Warehouse SaaS database the same way as your local applications connect to your IBM Db2 Warehouse SaaS database.

When your apps use the IBM Cloud platform, you can take advantage of the `VCAP_SERVICES` environment variable to simplify the task of specifying database details and credentials:

1. On your IBM Cloud dashboard, in the **Connections** tab of the Service Details page for your IBM Db2 Warehouse SaaS service, click the **Create connection** button.
2. Select the IBM Cloud app to use with your IBM Db2 Warehouse SaaS database as a data source, and then click the **Connect** button.
3. Update your application code to retrieve database details and credentials from the `VCAP_SERVICES` environment variable:

Example without `VCAP_SERVICES`

```
$ <?php
$driver      = "DRIVER={IBM DB2 ODBC DRIVER}";

$database    = "BLUDB";          # Get these database details from
$hostname    = "<Host-name>";    # the Connection info page of the
$port        = 50001;           # web console.
$user        = "<User-ID >";    #
$password    = "<Password>";    #
$dsn         = "DATABASE=$database;" .
              "HOSTNAME=$hostname;" .
              "PORT=$port;" .
              "PROTOCOL=TCPIP;" .
              "UID=$user;" .
              "PWD=$password;";

$conn_string = $driver . $dsn;

$conn        = db2_connect( $conn_string, "", "" );
?>
```

Example with `VCAP_SERVICES`

```
$ <?php
$driver      = "DRIVER={IBM DB2 ODBC DRIVER}";

$vcap        = json_decode( getenv( "VCAP_SERVICES" ), true );
$dsn         = $vcap[ "dashDB" ][0][ "credentials" ][ "dsn" ];

$conn_string = $driver . $dsn;

$conn        = db2_connect( $conn_string, "", "" );
?>
```

Database design best practices

IBM® Db2® Warehouse as a Service is a managed cloud data warehouse service that is preconfigured for optimal performance. The following section provides best practices for IBM Db2 Warehouse SaaS database design.

Design tips

- In IBM Db2 Warehouse SaaS, all tables are column organized by default for optimal performance on analytical workloads. While you can create row-organized tables, it is strongly recommended that you create all tables as column organized for good performance on analytical queries.
- As a SaaS offering, IBM Db2 Warehouse SaaS automatically manages table spaces, buffer pools, and database partition groups to ensure optimal performance. When you create your IBM Db2 Warehouse SaaS instance, you'll find the following database objects created by default:
 - One regular table space: `USERSPACE1` (32K page size)
 - One temporary table space: `TEMPSPACE1` (32K page size)
 - One buffer pool: `IBMDEFAULTBP` (32K page size)
 - One database partition group: `IBMDEFAULTGROUP`

It is strongly recommended to not create additional table spaces, temporary table spaces, buffer pools, or database partition groups.

- If you're using any one of the IBM Db2 Warehouse SaaS [multi-partitioned plans](#), you can optimize performance by defining a good set of table distribution keys. A good distribution key distributes data evenly across database partitions so that all of your database partitions can help process data for your queries. A poorly-chosen distribution key results in data skew. This means that data is stored unevenly across database partitions such that only a subset of data partitions participate in the processing while the rest remain idle. For more information about distribution keys, see [Distribution keys for multi-partitioned plans](#).



Note: If you're using a single-partitioned plan, there's no need to worry about distribution keys because all of the data is stored on one data partition.

- The Db2 engine includes many registry and environment variables, and database configuration parameters. As a SaaS offering, IBM Db2 Warehouse SaaS is deployed with preconfigured variables and parameters to optimize performance. Changing any of these settings might result in adverse effects on database operation or performance and is therefore strongly discouraged. If you think your system might benefit from changes to some registry variables, environment variables, or database configuration parameters, open a support case to request guidance from our service experts.

Advanced administration with IBM Db2 Data Management Console

Every IBM® Db2® Warehouse as a Service instance comes with a built-in console. This console provides the ability to load data into your IBM Db2 Warehouse SaaS instance, run SQL, and monitor core metrics for your instance, including resource usage, responsiveness, and throughput. If you need more advanced capabilities, including monitoring multiple data warehouses at once, then we suggest connecting your IBM Db2 Warehouse SaaS instance to [IBM Db2 Data Management Console](#).

By connecting IBM Db2 Warehouse SaaS to Db2 Data Management Console, you can access advanced diagnostic real time and historical monitoring, statement tracking and optimization, application locking details, storage, memory, and over 20 other KPIs for your IBM® Db2® Warehouse as a Service instance.

With Db2 Data Management Console, numerous alerts for performance optimization and availability are also available for around-the-clock automated monitoring of your IBM Db2 Warehouse SaaS instance.

Scaling

IBM® Db2® Warehouse as a Service provides you with the ability to independently scale compute and storage. You can use the console to scale resources immediately or to schedule a scaling operation, or you can scale your instance programmatically using the REST API.

To use the console, click **Administration** and then click the **Scaling** tab on the **Compute & storage** page.

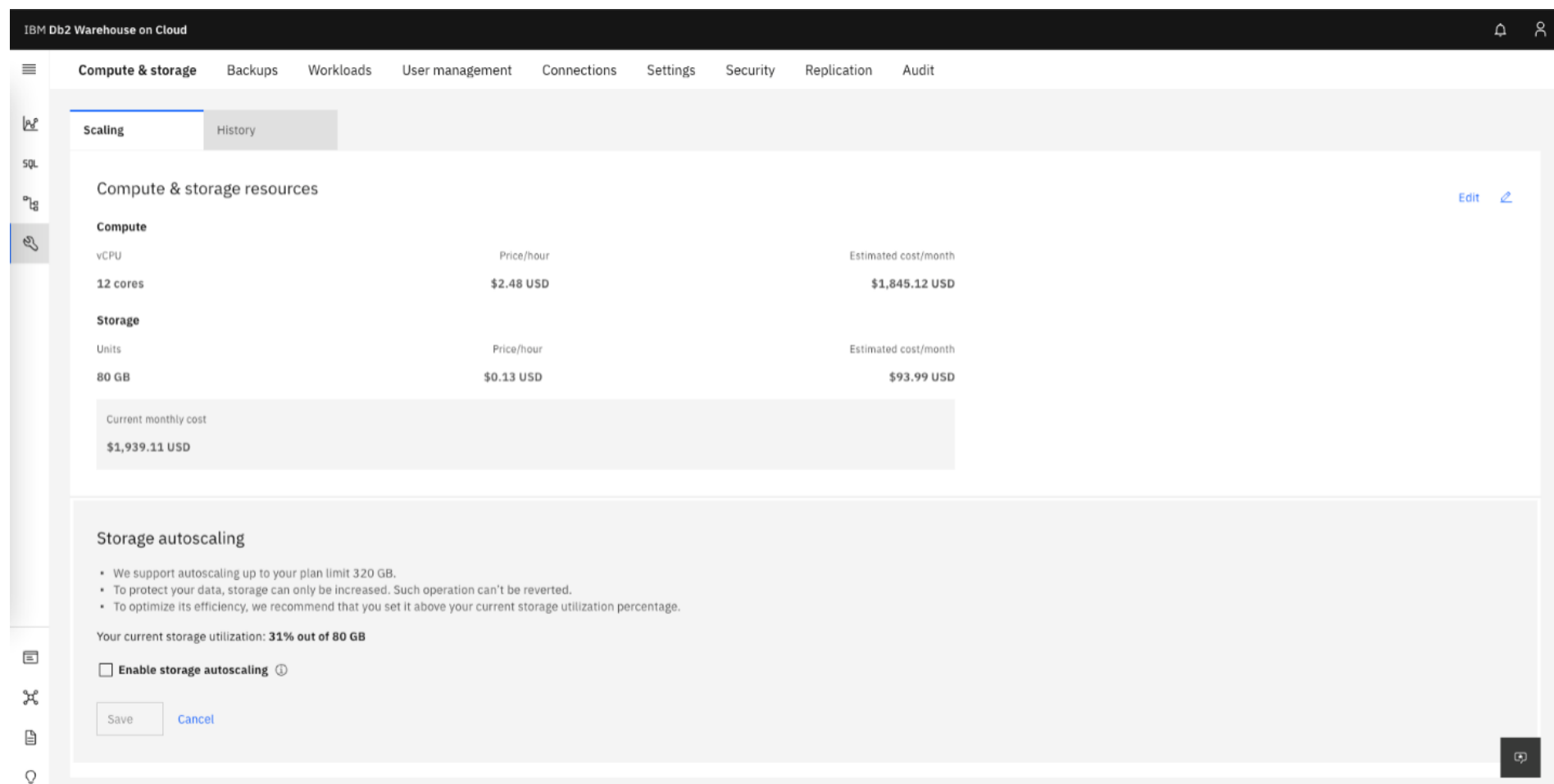


Figure 1. A screenshot of the Scaling tab

For information about using the REST API to scale compute or storage, see [REST API](#).

On-demand scaling is charged on an hourly basis. This allows you to pay for additional resources only when they are needed. For example, you can scale up compute if your workload is heavier on weekdays, and then scale it down for the weekend.

Compute Scaling

With all plans, you can scale compute up or down. Memory is scaled up or down proportionally as you scale compute resources up or down. During compute scaling, all uncommitted transactions are rolled back.

Approximately 45 minutes of downtime is required for a compute scale operation. The downtime will vary depending on the amount of crash recovery processing that might be required. The best practice is to gracefully end workloads prior to a scaling operation, or to schedule the scaling operation during a low period of activity.

Storage Scaling

With the current generation of plans hosted on AWS (plans deployed in or after July 2023 that support native object storage), you can scale storage up or down. The ability to scale storage down allows you to reduce your costs as you move data from more expensive block storage to less expensive cloud storage. With all other plans, you can only scale storage up.

You can also enable auto-scaling for your storage. When you enable autoscaling, the storage on your Db2 Warehouse on Cloud instance will automatically be scaled up if your storage use exceeds the threshold you specify. The following screenshot shows the auto-scaling panel in the console for a Db2 Warehouse on Cloud Flex instance hosted on IBM Cloud.

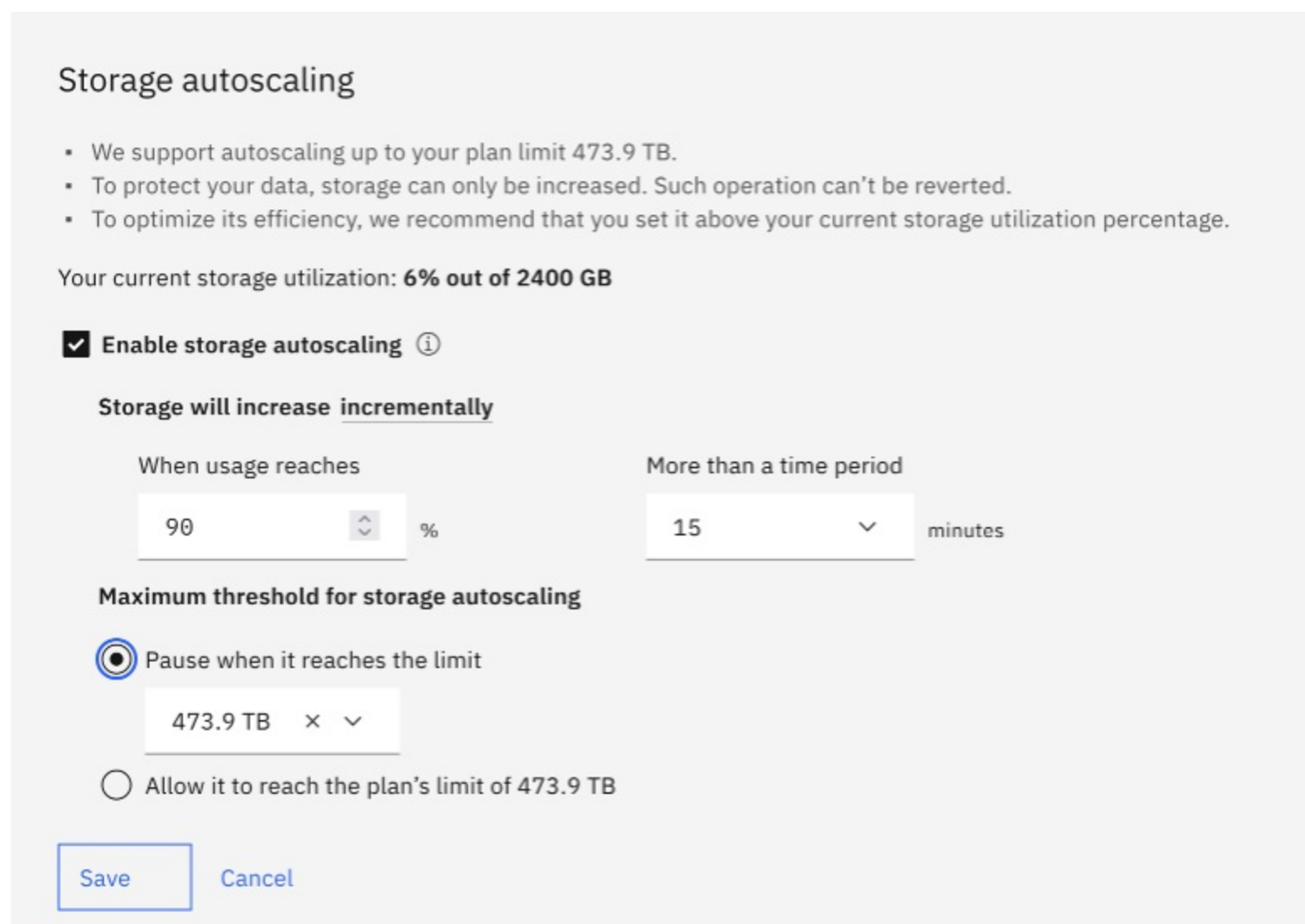


Figure 2. A screenshot of the autoscaling tab

Scaling storage up is an online operation. Scaling storage down requires downtime.

Backup and restore

A snapshot backup of the database is taken daily. Management and configuration of daily snapshot backups are built into the web console.

You can use the web console to restore from a snapshot backup if needed. While the restore is in progress, all writes in the system are queued, and all reads that don't depend on the queued writes will continue.

IBM Cloud

The last 7 daily snapshots are retained by default. Snapshot backups are encrypted and stored in block storage local to the Db2 Warehouse on Cloud system. Snapshot backups are free of charge.

Amazon Web Services

The last 7 daily snapshots are retained by default. When deployed on Amazon Web Services, you can use the web console to configure a longer retention period for snapshot backups if desired.

Potentially unlimited snapshots can be retained. Snapshot backups are encrypted and stored in Amazon Web services S3. S3 keeps copies of each snapshot backup across 3 availability zones (AZs) in each region by default, so there are 3 copies of each snapshot backup in total.

With the previous generation of plans, the first 7 snapshot backups on Amazon Web Services are free of charge. You will be charged each month for the

capacity required for any additional snapshot backups.

With the current generation of plans, you are charged for all backups. The backup process backs up data on both block and object storage. The backup in this case includes snapshots of block storage and AWS S3 backup of object storage data.

Cloud provider	Backup frequency	Number of retained backups	Retention period
IBM Cloud	1 / day	Up to 7	7 days; FIFO* rollover
Amazon Web Services (previous generation)	1 / day	Up to 7 by default; Can be increased	7 days; FIFO* rollover
Amazon Web Services (current generation)	1 / day	7 by default; can be increased or decreased by configuring retention period	7 days default; can be increased or decreased; FIFO rollover

Table 1. Snapshot backup frequency and retention period

*First in, first out

Logical schema backup and restore

This feature provides the ability to do full, cumulative incremental, or delta incremental backup of a Db2 schema followed by full restore of the schema or table(s) within the schema. Logical schema backup is a flexible and lightweight way to backup and restore table level data. For more information about this feature, see [Schema-level and table-level backup and restore](#).

Disaster recovery

Disaster recovery (DR) backups for IBM® Db2® Warehouse as a Service are enabled by default.

If a disaster event occurs at the data center where your Db2 Warehouse on Cloud instance is deployed, IBM service operators will work with you to stand up a new data warehouse in a different data center, by using the most recent disaster recovery backup.

For more information about DR and replication on Db2 Warehouse SaaS, see [Replication](#).

IBM Cloud

Current Generation

When deployed on IBM Cloud, the daily snapshot backup described in <https://cloud.ibm.com/docs/Db2whc?topic=Db2whc-br> is also used for disaster recovery purposes. This DR backup is encrypted and stored in IBM Cloud Object Storage (COS) across 3 availability zones (AZs) in each region. Regional DR backups allow you to restore your instance to another zone in the same region if needed. The RPO for DR backups on IBM Cloud is 24 hours. The RTO if a disaster occurs will be best effort. The actual time required to complete a DR restore is dependent on many variables, including but not limited to:

- the number of vCPUs in the instance
- the amount of data stored in Native Cloud Object Storage (<https://www.ibm.com/docs/en/db2w-as-a-service?topic=native-cloud-object-storage-support>)

With the current generation of plans, the console allows you to copy backups to a separate region, and restore a backup to an instance in another region if needed. The region must be a supported region for Db2 Warehouse SaaS on IBM Cloud. You are charged for the storage used for these copies, and for data transfer charges.

Previous Generation

For instances deployed on IBM Cloud, disaster recovery (DR) backups supplement daily snapshot backups. They are used exclusively for system recovery purposes by IBM service operators if there is a disaster or system loss.

A full backup of the database is taken once a week for disaster recovery. This DR backup is encrypted and stored in IBM Cloud Object Storage (COS). IBM COS replicates each DR backup across multiple IBM Cloud regions to ensure availability if a single zone fails. You can open a support ticket to ask which regions your backups are being replicated to.

DR backups of the last 2 weeks are retained by default. The RPO for DR backups on IBM Cloud is 1 week. The RTO if a disaster occurs is dependent upon the size of the database – 1.5 hours per terabyte of data.

It is essential for disaster recovery backups to be taken regularly as part of this managed service. If you have concerns about the time window in which backups run and want to schedule the backup for a less busy time on your system, open a support case.

Amazon Web Services

Current Generation

When deployed on Amazon Web Services, the daily snapshot backup described in <https://cloud.ibm.com/docs/Db2whc?topic=Db2whc-br> is also used for disaster recovery purposes. This DR backup is encrypted and stored in Amazon Web Services S3 across 3 availability zones (AZs) in each region. Regional DR backups allow you to restore your instance to another zone in the same region if needed. The RPO for DR backups on Amazon Web Services is 24 hours. The RTO if a disaster occurs will be best effort. The actual time required to complete a DR restore is dependent on many variables, including but not limited to:

- the number of vCPUs in the instance
- the amount of data stored in Native Cloud Object Storage (<https://www.ibm.com/docs/en/db2w-as-a-service?topic=native-cloud-object-storage-support>)

With the current generation of plans, the console allows you to copy backups to a separate region, and restore a backup to an instance in another region if needed. The region must be a supported region for Db2 Warehouse SaaS on Amazon Web Services. You are charged for the storage used for these copies, and for data transfer charges.

Previous Generation

When deployed on AWS, the daily snapshot backup described in <https://cloud.ibm.com/docs/Db2whc?topic=Db2whc-br> is also used for disaster recovery. This DR backup is encrypted and stored in Amazon Web Services S3 across 3 Availability Zones (AZs) in each region. Regional DR backups allow you to restore your instance to another zone in the same region if needed. The RPO for DR backups on Amazon Web Services is 24 hours. The RTO if a disaster occurs is approximately 4 hours.

Brazil: Supplementary Rule 14 (applies to systems provisioned for the Brazilian federal government)

For the current generation of plans hosted on AWS, the disaster recovery (DR) option for IBM Db2 Warehouse SaaS offerings is not available in Brazil for the federal government due to Supplementary Rule 14.

High availability (HA)

The measure of a successful database solution is its availability. The architecture of IBM® Db2® Warehouse as a Service features multiple tiers of reliability, which allows for fast, automated recovery from any issues that the data warehouse might encounter.

Compute HA

Any node failure is immediately detected by the cloud provider's container service. The containers and pods that were running in the failed node are scheduled to a new node from a pool of nodes. The system is back to 100% normal operation after a short downtime.

Storage HA

IBM Db2 Warehouse SaaS uses highly-reliable block storage, designed to provide consistent performance and protect data integrity and availability through maintenance events and unplanned failures. Storage is provisioned and managed independently of compute.

Network HA

Network connections are made highly available by provisioning your service with a redundant network interface card (NIC).

If the container service detects a network issue, pods and containers can automatically restart after a short downtime.

Diagnosing and resolving high storage usage

This document describes ways to diagnose and, if necessary, reduce the storage usage on your IBM Db2 Warehouse SaaS system.

Check storage usage

Run the following SQL query as an `admin` user to check the current storage usage on every Db2 partition in your IBM Db2 Warehouse SaaS instance:

```
db2 "SELECT DBPARTITIONNUM , ROUND((SUM(FS_USED_SIZE) / SUM(FS_TOTAL_SIZE)::DECFLOAT) * 100,2) AS UTILIZATION_PERCENT, ROUND((SUM(FS_TOTAL_SIZE) - SUM(FS_USED_SIZE)::DECFLOAT) / 1073741824,2) AS FREE_GB FROM TABLE(ADMIN_GET_STORAGE_PATHS(NULL,-2)) AS T WHERE STORAGE_GROUP_NAME NOT LIKE '%TEMP%' GROUP BY DBPARTITIONNUM ,CASE WHEN DBPARTITIONNUM = 0 THEN 'NP' ELSE 'P' END"
```

There are two possible causes for high storage usage:

- The storage attached to your system is reaching maximum capacity.
- There is empty space on the disks that must be reclaimed.

Diagnosing and resolving

If the output from the previous SQL query indicates that storage usage on your instance is high, the following steps will help to diagnose and resolve this scenario:


1. Check if the instance has reclaimable space by running the following SQL query as an `admin` user:

```
db2 "SELECT SUBSTR(TBSP_NAME, 1,30) AS TBSP_NAME, SUM(TBSP_USED_PAGES * TBSP_PAGE_SIZE)/ 1073741824 AS TBSP_USED_SIZE_GB,
SUM((TBSP_PENDING_FREE_PAGES+ tbspc_free_pages) * TBSP_PAGE_SIZE)/ 1073741824 AS FREE_OR_PENDING_GB FROM
TABLE(MON_GET_TABLESPACE(' ', -2)) where TBSP_NAME not like '%SYS%' and TBSP_NAME not like '%TEMP%' and
RECLAIMABLE_SPACE_ENABLED = 1 and (TBSP_PENDING_FREE_PAGES+ tbspc_free_pages) >= 1 GROUP BY TBSP_NAME ORDER BY 3 DESC"
```


1. If the data is very skewed across Db2 partitions, meaning that one or more Db2 partitions is almost full while other Db2 partitions still have significant capacity remaining, then see [Choosing a hash distribution key for a table in an MPP database](https://www.ibm.com/support/knowledgecenter/SS6NHC/com.ibm.swg.im.dashdb.doc/learn_how/choosing_dist_key_mpp.html) { : external }.


```

2. If one or more table spaces have reclaimable space, complete the following steps to reclaim that space:

 **Important:** Do not reclaim space during an online backup. The reclaim operation can also impact the performance of ongoing workloads, so it is best to run the following steps during off-peak hours.

- a. Verify that an online backup is not running:

```
db2 "SELECT MEMBER, SUBSTR(UTILITY_DETAIL, 1, 150) AS CMD FROM TABLE(MON_GET_UTILITY(-2)) AS T"
```


b. For every table space in the list generated by the previous step, reclaim the unused space by running the following command:



```
{: codeblock}
db2 alter tablespace <tablespace_name> reduce max
```


```

3. If there is no disk space that can be reclaimed, you must either purge unwanted data or scale up the storage on your IBM Db2 Warehouse SaaS instance to ensure database stability.

Additional information

[Reclaimable storage](#)

[Reclaiming unused storage in DMS table spaces](#)

[ALTER TABLESPACE statement](#)

[Distribution keys for multi-partitioned plans](#)

Maintenance windows

Routine maintenance windows are scheduled to ensure that your IBM® Db2® Warehouse as a Service system receives the latest features, patches, and upgrades. During these maintenance windows, updates are applied automatically without your intervention. For posts of maintenance windows, see [IBM Cloud status: Planned Maintenance](#). We contact you directly if your system specifically needs maintenance.

If you have business situations or conditions that might be impacted by the scheduling of our maintenance windows, open a [support case](#) to inform us about your concerns.

Data virtualization (federation)

Db2 data virtualization (also known as federation) is supported by IBM Db2 Warehouse SaaS. Data virtualization gives you single-query access to all of your data that is on multiple distributed databases anywhere in your organization. You can access data that is on any of your Db2 or Informix data sources, both in the cloud and on premises.

This function is supported on all versions of IBM Db2 Warehouse SaaS.

Use Cases

Consolidate data sources

By federating your data sources that are located both in the cloud and on premises anywhere in your organization, your virtualized data appears to be retrieved from a single source. Data virtualization eliminates the burdensome and costly data migration process, giving you the ability to analyze all of your data efficiently and cost effectively.

Attach to Db2 on Cloud

Users of products in the Db2 family can federate data from IBM Db2 SaaS and IBM Db2 Warehouse SaaS databases. From a common interface to access the data, you can easily add, query, and analyze your data without complex ETL processes and without any additional code.

Increase database capacity beyond fixed limits

Federation gives you the ability to increase the capacity of an on-premises database by federating with a database on the cloud. Data virtualization in this case is a great option if your on-premises database is running out of storage. Increasing the capacity of a database with federation is useful for new development because developers don't need to change a database already in production. You can also federate between two IBM Db2 Warehouse SaaS databases to increase the database capacity beyond the current limits of the Flex plan.

Getting started

The following steps are an example of how you go about federating your disparate data sources to appear as if the data is retrieved from a single source. The following example illustrates the federation of two IBM Db2 Warehouse SaaS databases:

On the Db2 Warehouse on Cloud target machine

Host name: targetdotcom

1. Create a table `testdata` in schema `admin2`.
2. From the IBM Db2 Warehouse SaaS console, load the `testdata` table with data as user `admin2` with password `YYYY`.

On a Db2 Warehouse on Cloud machine being used as a federation source

From the IBM Db2 Warehouse SaaS console:

1. Create a server to talk to the target machine:

```
create server <server_name> type dashdb version 11 wrapper drda authorization "<admin_user_on_target>" password "  
<admin_password_on_target>" options (host '<target_host_name>', port '50000', dbname 'bludb')
```

For example:

```
create server db2server type dashdb version 11 wrapper drda authorization "admin2" password "YYYY" options (host  
'targetdotcom', port '50000', dbname 'bludb')
```

2. Create the user mapping for admin2:

```
create user mapping for <admin_user> server db2server options (remote_authid '<admin_user_on_target>', remote_password  
'<admin_password_on_target>')
```

For example:

```
create user mapping for admin1 server db2server options (remote_authid 'admin2', remote_password 'YYYY')
```

3. Create a nickname for the database:

```
create nickname <nickname> for <server_name>.<schema_name>.<table_name>
```

For example:

```
create nickname ntest1 for db2server.admin2.testdata
```

4. Test that you can pull data from the target server:

```
select * from <nickname>
```

For example:

```
select * from ntest1
```

Additional information

For more information about data virtualization (federation), see: [Federation](#).

For information about the data sources supported by federation, see: [Federation Supported Data Sources](#).

Changing your Db2 Warehouse entitlements to a Cloud Pak for Data SaaS Subscription

Migrating your stand-alone parts to a Cloud Pak for Data subscription gives you more flexibility to choose which offerings you want to consume, and how

much you want to consume.

A subscription allows you to buy credits that you can use as quickly or slowly as you want to, based on your changing needs. If you want to provision a new instance or scale up storage or compute for an existing instance, you can initiate the request within minutes, without the need for a charge agreement to be created or amended. A subscription also provides you with the opportunity to explore a larger portfolio of offerings.

Steps

1. Purchase an IBM Cloud Pak for Data as-a-Service subscription.
2. Apply this subscription to the IBM Cloud account that currently contains your Db2 Warehouse on Cloud instance.
3. Create a case with support (<https://cloud.ibm.com/unifiedsupport/supportcenter>) and request to have your instance converted to use this subscription.

Security

Data security & encryption

The IBM Db2 Warehouse SaaS service has security built into all levels of its architecture.

The following methods are used to secure your data:

- Data at rest and database backups are encrypted using NIST SP 800-131A compliant cryptographic algorithms
- Data in motion is encrypted through SSL/TLS
- When deployed to IBM Cloud, backplane network connectivity is supported through IBM Cloud Service Endpoints
- When deployed to Amazon Web Services, backplane network connectivity is supported through Amazon Web Services PrivateLink
- Database-level security is supported through Role-Based Access Control (RBAC) and Row and Column Access Control (RCAC)
- Inter-node encryption is always enabled for the current generation of plans hosted on AWS, where available. It can optionally be enabled for your Flex or Flex Performance instance hosted on IBM Cloud. For more information, see [Inter-node encryption](#)

Encrypted connections are enforced by default. For more information, see [SSL connectivity](#).

Inter-node encryption

For the current generation of plans hosted on AWS, inter-node encryption is always enabled, where available. As of September 2023, inter-node encryption is available in all AWS regions except sa-east-1 (Sao Paulo). For the Flex and Flex Performance plans hosted on IBM Cloud, you can optionally choose to enable this form of encryption.

Inter-node encryption is an additional layer of security that protects inter-node traffic between internal Db2 nodes in a massively parallel processing (MPP) cluster. It supplements the SSL encryption that is already in use between the server and your applications.

Inter-node encryption ensures that your data is processed with encryption at all points from your application to physical storage and back, which protects your information from physical and software-based attacks. Leaving inter-node traffic unencrypted means that your data is transferred in plain text between the physical hardware nodes that make up your Db2 MPP instance. IBM already employs a variety of security protocols to protect such traffic to ensure the security and integrity of the unencrypted internal data flows. With this new feature, IBM provides an additional security layer to protect the data.

Due to the additional processing required to encrypt data on send and decrypt on receive, workloads with significant data movement may see a performance impact as little as 5% or as much as 20%. If your instance is hosted on IBM Cloud, your solution architect must evaluate this performance impact against your need to encrypt internal traffic flows. In some cases, it might be warranted (for instance in heavily regulated industries or where end users demand such encryption). In other cases, the existing in-depth security protocols may be sufficient.

IBM Cloud plans only: To enable inter-node encryption, open the web console, select **Administration**, and navigate to the **Settings** -> **Encryption** tab. Enabling or disabling inter-node encryption can be done online, and takes effect immediately. Choosing to enable inter-node encryption does not impact the Service Level Agreement (SLA), self-service backups, database replication, or scaling.

Identity and access management (IAM) on IBM Cloud

Identity and access management (IAM) enables you to securely authenticate users for platform services and control access to resources consistently across the IBM Cloud platform. For example, with only a single login to IBM Cloud with your IBMid, you have access to any of your service consoles and their applications without having to log in to each of them separately.

Features of IBM Cloud IAM

The following IAM features are implemented for the IBM Db2 Warehouse SaaS managed service with two types of supported identities:

IBMid

Users with an IBMid must be added to each database service instance by the database administrator through the console or REST API before these users can connect to the particular database service instance. Like for a non-IBMid user, a user ID for the database service instance must be entered at the same time that the IBMid user is added. This user ID needs to be unique within the database service instance. This user ID is also the authorization (AUTH) ID within the database. The database administrator can grant and revoke permissions based on the AUTH ID.

Service IDs

A service ID identifies a service or application similar to how a user ID identifies a user. The service IDs are IDs that can be used by applications to authenticate with an IBM Cloud service. A service ID represents a separate entity from the owning IBMid. Therefore, different authorities and permissions can be granted specific to the service ID within the database. Service IDs do not have passwords. An API key must be created for each service ID for the service ID to connect to the database service instance. For more information about service IDs, see: [Introducing IBM Cloud IAM Service IDs and API Keys](#)

and [Increase Information Security for Db2 on IBM Cloud](#).

Roles and actions

Every user that accesses the IBM Db2 Warehouse SaaS service in your account must be assigned an access policy with an IAM role. The access policy that you assign to users in your account determines what actions a user can perform within the context of the service or specific instance that you select. The allowable actions are customized and defined by IBM Db2 Warehouse SaaS as operations that are allowed to be performed on the service. Each action is mapped to an IAM platform or service role that you can assign to a user. If a specific role and its actions don't fit the use case that you're looking to address, you can [create a custom role](#) and pick the actions to include.



Tip: For information about the exact actions mapped to each role, see [IAM roles and actions](#) and [Db2 Warehouse](#).

Prerequisites

- Db2 Client V11.1 FP3 or later.
- Must use SSL connections through port 50001.
- [Configure your Db2 client](#).
- Catalog your database. See [Connecting to your database: step 2](#).

Client connections and user logins

The following methods can be used for IAM authentication:

Access token

An access token can be obtained from the IAM service directly by the application through the REST API by using an API key. For more information, see: [Getting an IBM Cloud IAM token by using an API key](#). The access token has a default validity period of 60 minutes before it expires. If the token has expired, the Db2 server won't allow the connection to be established. The token isn't checked for expiry after the connection is established. Just as it was before IAM integration, the connection stays connected until the application disconnects or the connection is terminated due to other reasons.

```
curl -k -X POST \  
  --header "Content-Type: application/x-www-form-urlencoded" \  
  --header "Accept: application/json" \  
  --data-urlencode "grant_type=urn:ibm:params:oauth:grant-type:apikey" \  
  --data-urlencode "apikey=<apikey>" \  
  "https://iam.cloud.ibm.com/identity/token"
```

An access token identifies an IBMid user or a service ID to the database. The database server verifies the validity of the access token before accepting it. This is the best method if the application needs to connect to more than one database service instance or other IBM Cloud service instances because it minimizes the communication to the IAM service to validate the access token.

API key

Multiple API keys can be created for each IBMid user or service ID. Each API key is typically created for a single application. It allows the application to connect to the database service instance as long as the owning IBMid or service ID is added as a user to the same database service instance. The API key has the same authorities and permissions within the database as the owning IBMid or service ID. If an application should no longer be allowed to connect to the database, the corresponding API key can be removed. This method of authentication requires fewer changes in the application than using an access token as it requires no direct interaction with the IAM service. For more information about creating and managing API keys, see: [Managing user API keys](#).

IBMid/password

The IBMid/password can be used to log in to the console and can also be used within the application in the same ways that a user ID/password is allowed. The exception occurs when the IBMid is federated because a redirection to your organization's login page cannot be done. However, the recommended method for an application to establish a connection to the database service instance is to use an access token.

Supported interfaces

The following database client interfaces are supported:

- [ODBC](#)
- [CLP](#)
- [CLPPLUS](#)
- [JDBC](#)

ODBC, CLP, and CLPPLUS

For an ODBC application or a command-line client (CLP, CLPPLUS) to connect to a Db2 server by using IAM authentication, a data source name (DSN) needs to be configured first in a `db2dsdriver.cfg` configuration file by running the following command:

```
db2cli writecfg add -dsn <dsn_alias> -database <database_name> -host <host_name_or_IP_address> -port 50001 -parameter "Authentication=GSSPLUGIN;SecurityTransportMode=SSL"
```

The `db2dsdriver.cfg` configuration file is an XML file, typically located in the `sqllib/cfg` directory, that contains a list of DSN aliases and their properties.

The following example of a `db2dsdriver.cfg` configuration file shows the configurations that are used to establish a connection to a database service instance. The configuration file provides the DSN alias, the database name, the host name (or IP address), and the **Authentication** type and **SecurityTransportMode** parameter values:

```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
<configuration>
  <dsncollection>
    <dsn alias="<data_source_name>" name="bludb" host="<host_name_or_IP_address>" port="50001">
      <parameter name="Authentication" value="GSSPLUGIN"/>
      <parameter name="SecurityTransportMode" value="SSL"/>
    </dsn>
  </dsncollection>
  <databases>
    <database name="bludb" host="<host_name_or_IP_address>" port="50001"/>
  </databases>
</configuration>
```

ODBC

The ODBC connection string can contain one of the following:

```
$ - **Access token**

`DSN=<dsn>;ACCESSTOKEN=<access_token>`

- **API key**

`DSN=<dsn>;APIKEY=<api_key>`

- **IBMid/password**

`DSN=<dsn>;UID=<ibmid>;PWD=<password>`
```

For ODBC, the **AUTHENTICATION=GSSPLUGIN** can be specified in either the `db2dsdriver.cfg` configuration file or in the application's connection string.

CLP

The CLP CONNECT statement can contain one of the following:

```
$ - **Access token**

Connect to the database server `<database_server_name>` and pass the access token by running the following command at the CLP command prompt or script:

`CONNECT TO <database_server_name> ACCESSTOKEN <access_token_string>`

- **API key**

Connect to the database server `<database_server_name>` with an API key by running the following command at the CLP command prompt or script:

`CONNECT TO <database_server_name> APIKEY <api-key-string>`

- **IBMid/password**

Connect to the database server `<database_server_name>` with an IBMid/password by running the following command at the CLP
```

command prompt or script:

```
`CONNECT TO <database_server_name> USER <IBMid> USING <password>`
```

For more details about connecting to a database server with CLP, see: [CONNECT \(type 2\) statement](#).

CLPPLUS

The CLPPLUS CONNECT statement can contain one of the following:

\$ - ****Access token****

Connect to the DSN alias (`@<data_source_name>`) and pass the access token by running the following command at the CLPPLUS command prompt or script:

```
`connect @<data_source_name> using(accesstoken <access_token_string>)`
```

- ****API key****

Connect to the DSN alias (`@<data_source_name>`) with an API key by running the following command at the CLPPLUS command prompt or script:

```
`connect @<data_source_name> using(apikey <api-key-string>)`
```

- ****IBMid/password****

Connect to the DSN alias (`@<data_source_name>`) with an IBMid/password by running the following command at the CLPPLUS command prompt or script:

```
`connect <IBMid>/<password>@<data_source_name>`
```

For more details about connecting to DSN aliases with CLPPLUS, see: [DSN aliases in CLPPlus](#).

JDBC

Type 4 JDBC Driver is supported for IAM authentication.

The following examples show connection snippets for the three methods:

- **Access token**

```
DB2SimpleDataSource dataSource;  
  
dataSource.setDriverType( 4 );  
dataSource.setDatabaseName( "BLUDB" );  
dataSource.setServerName( "<host_name_or_IP_address>" );  
dataSource.setPortNumber( 50001 );  
dataSource.setSecurityMechanism( com.ibm.db2.jcc.DB2BaseDataSource.PLUGIN_SECURITY );  
dataSource.setPluginName( "IBMIAMauth" );  
dataSource.setAccessToken( "<access_token>" );  
Connection conn = dataSource.getConnection( );
```

or

```
Connection conn = DriverManager.getConnection( "jdbc:db2://<host_name_or_IP_address>:50001/BLUDB:accessToken=  
<access_token>;securityMechanism=15;pluginName=IBMIAMauth;sslConnection=true" );
```

- **API key**

```
DB2SimpleDataSource dataSource;  
  
dataSource.setDriverType( 4 );  
dataSource.setDatabaseName( "BLUDB" );  
dataSource.setServerName( "<host_name_or_IP_address>" );  
dataSource.setPortNumber( 50001 );  
dataSource.setSecurityMechanism( com.ibm.db2.jcc.DB2BaseDataSource.PLUGIN_SECURITY );  
dataSource.setPluginName( "IBMIAMauth" );  
dataSource.setApiKey( "<api_key>" );  
Connection conn = dataSource.getConnection( );
```


or

```
Connection conn = DriverManager.getConnection( "jdbc:db2://<host_name_or_IP_address>:50001/BLUDB:apiKey=
<api_key>;securityMechanism=15;pluginName=IBMIAMauth;sslConnection=true" );
```

- **IBMid/password**

```
DB2SimpleDataSource dataSource;

dataSource.setDriverType( 4 );
dataSource.setDatabaseName( "BLUDB" );
dataSource.setServerName( "<host_name_or_IP_address>" );
dataSource.setPortNumber( 50001 );
dataSource.setSecurityMechanism( com.ibm.db2.jcc.DB2BaseDataSource.PLUGIN_SECURITY );
dataSource.setPluginName( "IBMIAMauth" );
Connection conn = dataSource.getConnection( "<IBMid>", "<password>" );
```

or

```
Connection conn = DriverManager.getConnection( "jdbc:db2://<host_name_or_IP_address>:50001/BLUDB:user=<IBMid>;password=
<password>;securityMechanism=15;pluginName=IBMIAMauth;sslConnection=true" );
```

Console user experience

The service console login page has the option to log in with your IBMid and password. After the **Sign In via IBMid** button is clicked, the user is directed to the IAM login page, on which the password is entered. When authentication is completed, the user is redirected back to the console. Before such login can be successful, the IBMid user must be added to each database service instance by the database administrator through the console or REST API. Like for a non-IBMid user, a user ID for the database service instance must be entered at the same time that the IBMid user is added. The user ID needs to be unique within the database service instance. This user ID is also the authorization (AUTH) ID within the database.

To add a user with either an IBMid or a service ID by using the web console, complete the following steps:

1. Log in as a user who has administrator privileges.
2. Select **Administration**, and navigate to **User Management**.

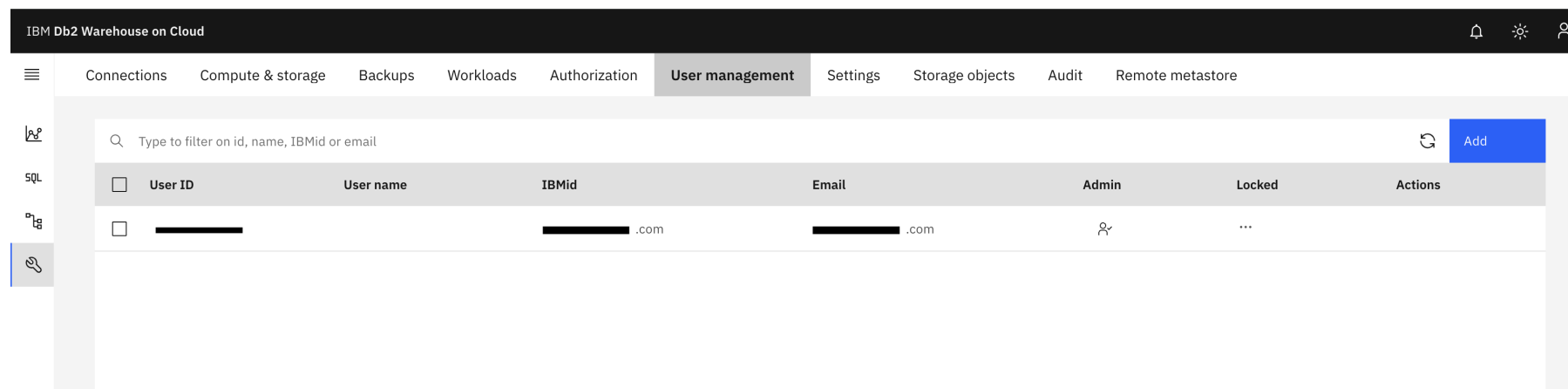


Figure 1. Selecting Manage Users page from drop-down menu

3. Click **Add**.

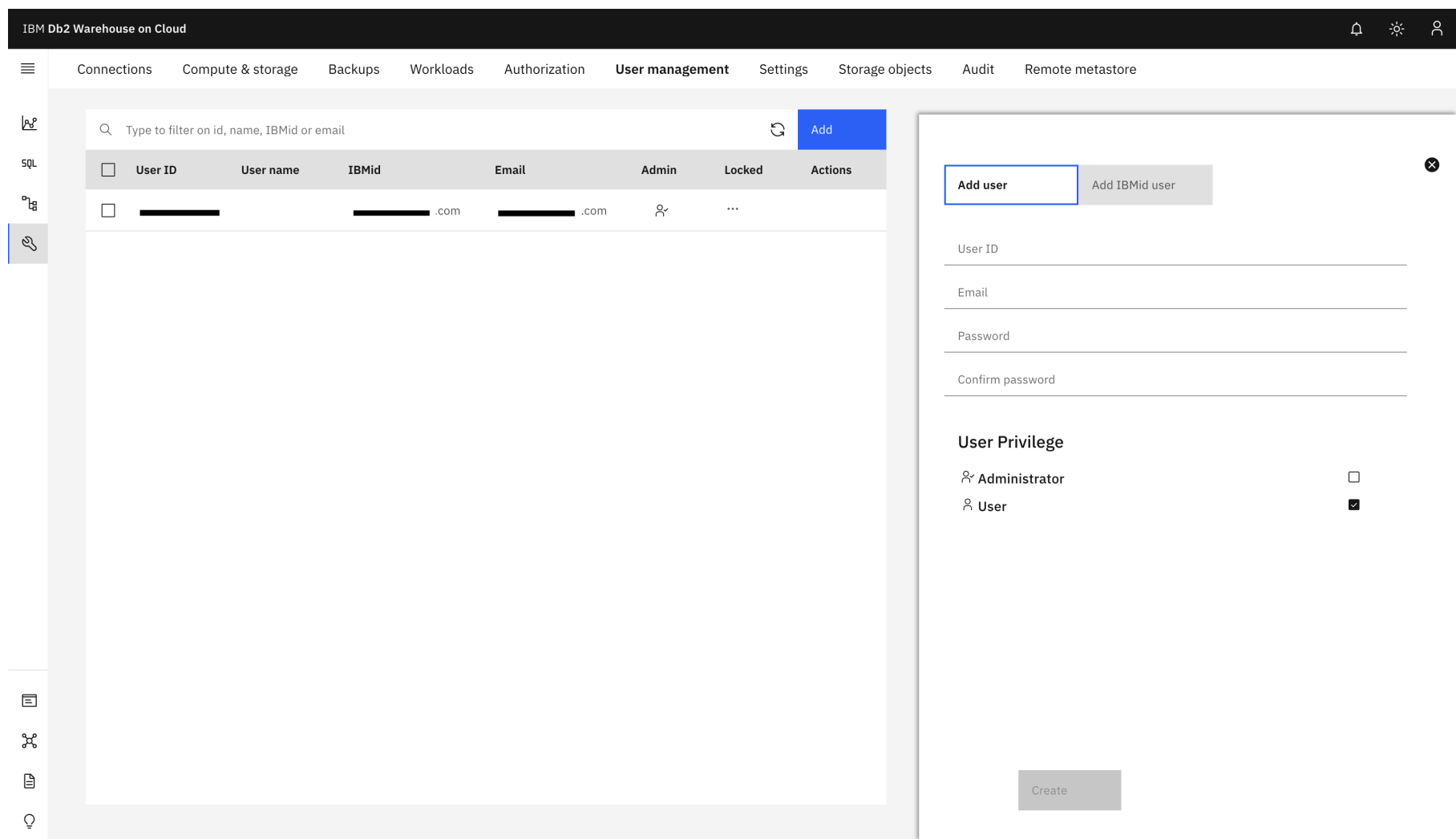



Figure 2. Clicking Add to create a user

4. Select **Add IBMid user**.





Add userAdd IBMid user

User ID

IBMid


User Privilege

 Administrator	<input type="checkbox"/>
 User	<input checked="" type="checkbox"/>

Create

Figure 3. Creating an IBMid user

5. Enter a **User ID** and an **IBMid** in the provided fields. Click **Create**.

 **Note:** The **IBMid** field can be used to specify either an IBMid or a service ID for the user.

REST API experience

The IBM Db2 Warehouse SaaS REST API was enhanced to also accept an IAM access token for the functions that previously accepted a database service-generated access token.

- To add a new IBMid user, run the following example API call:

```
curl --tlsv1.2 "https://<IPaddress>/dbapi/v3/users" -H "Authorization: Bearer <access_token>" -H "accept: application/json"
-H "Content-Type: application/json" -d '{"id": "<userid>", "ibmid":
"<userid>@<email_address_domain>", "role": "bluadmin", "locked": "no", "iam": true}'
```



Note: The userid value for `"id"` cannot match the full email ID. It can match the first part of the email ID, but that is not necessary. The two different IDs are not linked together in any way. For example, if your `"ibmid"` is `abc@us.ibm.com`, then you could specify `abc` as the ID, or you could specify `def` for example, but you cannot specify `abc@us.ibm.com`.

- To migrate an existing non-IBMid database user (for example, `abcuser`) and make them an IBMid user, first delete the non-IBMid user ID by running the following example API call:

```
curl --tlsv1.2 -X DELETE "https://<IPaddress>/dbapi/v3/users/abcuser" -H "Authorization: Bearer <access_token>" -H "accept:
application/json" -H "Content-Type: application/json"
```

Next, re-add the user with an IBMid that is the same as the previous user ID (`abcuser`) by running the following example API call:

```
curl --tlsv1.2 "https://<IPaddress>/dbapi/v3/users" -H "Authorization: Bearer <access_token>" -H "accept: application/json"
-H "Content-Type: application/json" -d "
{"id": "abcuser", "ibmid": "abcuser@<email_address_domain>", "role": "bluadmin", "locked": "no", "iam": true}'
```

Because the user ID (`abcuser`) remains the same for the new IBMid for that user, the granted database permissions for the user remain unchanged. If a list of existing non-IBMid database users needs to be migrated to make them IBMid users, you are required to run the previous pair of API calls for each user.

- To add many new IBMid users at one time, create a batch file that lists the following example API calls, one for each user:

```
curl --tlsv1.2 "https://<IPaddress>/dbapi/v3/users" -H "Authorization: Bearer <access_token>" -H "accept: application/json"
-H "Content-Type: application/json" -d '{"id": "<userid1>", "ibmid":
"<userid1>@<email_address_domain>", "role": "bluadmin", "locked": "no", "iam": true}'
curl --tlsv1.2 "https://<IPaddress>/dbapi/v3/users" -H "Authorization: Bearer <access_token>" -H "accept: application/json"
-H "Content-Type: application/json" -d '{"id": "<userid2>", "ibmid":
"<userid2>@<email_address_domain>", "role": "bluadmin", "locked": "no", "iam": true}'
.
.
.
```

For more details about your service's API, see: [IBM Db2 Warehouse SaaS REST API](#).

IBMid federation

To use your own identity provider such as LDAP, you must first federate your LDAP server with IBMid. For instructions about federating your LDAP server with IBMid, see: [IBMid Enterprise Federation Adoption Guide](#). After IBMid federation is completed and the allowed users are added to the database service instance by the database administrator, these users can log in to the console with their company user ID and password. Alternatively, these users can use an access token or API key that represents their user ID to connect to the database service instance through one of the supported database client interfaces.

Restrictions

The following restrictions are regarding IAM authentication:

- IAM authentication for a Db2 client that is connecting to a Db2 server is only supported over an SSL connection.
- IBMid federation is supported to allow custom user management portal or server to be used for authentication. Such support does not include any group federation.
- IAM authentication for database federation is not supported.
- Running IDA and user-defined extension (UDX) commands through CLPPlus are not supported.
- Type 2 JDBC Driver is not supported.

Managing users

Access to IBM® Db2® Warehouse as a Service service instances for users in your account is controlled by [identity and access management \(IAM\) on IBM Cloud](#) and database access is provided by standard access controls provided by the database.

For more information about IAM, see [What is IBM Cloud Identity and Access Management?](#).

User types

Database users

These are the users that are used to access the database. Traditionally, these are the OS users in a typical Db2 deployment, although, in the cloud, a user registry is used. Db2 understands these users as native to the database. The database privileges for the users can be granted or revoked as can roles that are created by the user.

Database users are not granted any service-level functions. For example, a database administrator who has access to the data does not have the ability to change the configuration of the system outside of the database privileges that they were given.

IAM users

IAM is only integrated with high-level service access, which governs privileges and operations available in the IBM Db2 Warehouse SaaS console and database. Access to the database by these IAM users is provided by allowing an IAM user or service ID access to a specific Db2 user, as mentioned earlier.

Roles and access

Users can use JDBC or any Db2 client to connect to their database. There are two ways that users can access the database:

- Use their database user name and password associated with their account
- Use the IAM token (or APIKey, which gets the token) that is mapped to the associated database user

IAM authentication is performed as the authentication mechanism. Permissions are not controlled by IAM. Permissions are controlled by database level privileges of the associated user.

Console access

Console access is controlled by IAM. An IAM user can be assigned access by the IAM interface to all Db2 service instances, all Db2 service instances in a resource group, or a specific service instance. Within these parameters, IAM users can be assigned platform and service-level access.

Role	User mgmt	SQL editor/tables	Monitoring info	Settings (includes scale, backup etc.)	Info panels
IAM - Platform - Viewer	No	No (unless mapped to Db2 user)	Yes	No	Yes
IAM - Platform - Operator	No	No (unless mapped to Db2 user)	Yes	Yes	Yes
IAM - Platform - Editor	No	No (unless mapped to Db2 user)	Yes	Yes	Yes
IAM - Platform - Administrator	Yes	No (unless mapped to Db2 user)	Yes	Yes	Yes
Non-IAM, but authenticate with JDBC	Only "Change password"	Yes	No	No	Yes

Table 1. Roles and console permissions

Service action mapping

Service action access is also controlled by IAM Roles. An IAM user can be assigned access by the IAM interface to all Db2 service instances, all Db2 service instances in a resource group, or a specific service instance. Within these parameters, IAM users can be assigned or revoked access from specific service actions.

Role	Manage-users	Scale	Restore	Settings	Backup	Monitor	View settings
IAM - Platform - Viewer	No	No	No	No	No	Yes	Yes
IAM - Platform - Operator	No	Yes	Yes	Yes	Yes	Yes	Yes

IAM - Platform - Editor	No	Yes	Yes	Yes	Yes	Yes	Yes
IAM - Platform - Administrator	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Table 2. Roles and service actions

For more information about user management, see [Database user management](#)

User roles

You can define user roles to manage data access at the user level on your IBM® Db2® Warehouse as a Service instance. Users can be added to roles that reflect their job functions and you can enforce data access policies based on their respective roles.

User roles can easily be implemented through a series of SQL statements. You can use the SQL editor of the web console. For example, you can create a role for your data science team by running the following SQL statement:

```
CREATE ROLE DATASCIENCE
```

Then, if you want to give your data scientists access to sales data, run the following statement:

```
GRANT SELECT ON TABLE SALESDATA TO ROLE DATASCIENCE
```

You can repeat the previous statement for all of the tables that you want your data science team to be able to access.

Next, add your data science team to the `DATASCIENCE` role. Assuming that you have two data scientists, Alice and Bob who exist as users of the database, you can then add them to the `DATASCIENCE` role by running the following SQL statement:

```
GRANT ROLE DATASCIENCE TO USER ALICE, USER BOB
```

Alice and Bob are now able to work on all of the data that is accessible to the `DATASCIENCE` role.

For more information about user roles in IBM Db2 Warehouse SaaS, see [User-defined user roles](#).

Row and column access control (RCAC)

IBM® Db2® Warehouse as a Service comes equipped with powerful user security features that can help you manage security at the row and column level. This fine-grained access control that is known as row and column access control (RCAC) in IBM Db2 Warehouse SaaS, works with [user roles](#).

RCAC can be used to ensure that your database users can access only the portions of database tables that are required for their work. For example, a hospital might have a policy that states a doctor can see the data of only their patients and not the patients of other doctors. In the following example of a patient data table,

Patient name	Chart	Doctor name
Alice	Aa BB cc	Smith
Bob	aA Bb CC	Chan
Charlie	AA BB CC	Morton

Table 1. Patient data table

Dr. Smith is restricted to seeing only Alice's data, as depicted in the following table view:

Patient name	Chart	Doctor name
Alice	Aa BB cc	Smith

Table 2. Dr. Smith's view of the patient data table

You can apply RCAC rules to your IBM Db2 Warehouse SaaS instance to enforce the hospital's policy.

For more information about RCAC, see [RCAC overview](#).

Security compliances

IBM® Db2® Warehouse as a Service is provided on secure IBM Cloud® and Amazon Web Services platforms. Db2 managed services comply with the specific security standards and auditing processes that are listed here and do not automatically inherit general IBM Cloud security compliances that are documented for IaaS, PaaS, and SaaS cloud offerings.

For details about each of the security compliances, see [Details about security compliance](#).

IBM Db2 Warehouse as a Service Flex offering plans	HIPAA	ISO	SOC 2 Type 2	GDPR	Privacy shield
Flex One on IBM Cloud	✓	✓	✓	✓	✓
Flex on IBM Cloud	✓	✓	✓	✓	✓
Flex Performance on IBM Cloud	✓	✓	✓	✓	✓
Flex on Amazon Web Services		✓		✓	✓
Flex Performance on Amazon Web Services		✓		✓	✓

Table 1. Summary of security compliances for the Flex service offering plans

Key Protect integration

The data that you store in IBM® Db2® Warehouse as a Service is encrypted by default using randomly generated keys. If you need to control the encryption keys, you can use [IBM Key Protect](#) to create, add, and manage encryption keys. Then, you can associate those keys with your Db2 Warehouse on Cloud deployment to encrypt your Db2 databases.

To get started, you need [Key Protect](#) provisioned on your IBM Cloud account.

Creating or adding a key in Key Protect

Navigate to your instance of Key Protect and [generate or enter a key](#).

Granting service authorization

Authorize Key Protect for use with Db2 Warehouse SaaS deployments:

1. Open your IBM Cloud dashboard.
2. From the menu bar, select **Manage > Access (IAM)**.
3. In the side navigation, select **Authorizations**. Click **Create**.
4. In the *Source service* menu, select the service of the deployment. For example, **Db2 Warehouse**.
5. In the *Source service instance* menu, select **All service instances**.
6. In the *Target service* menu, select **Key Protect**.
7. In the *Target service instance* menu, select the service instance to authorize.
8. Enable the **Reader** role. Click **Authorize**.

Using the Key Protect key

After you grant your Db2 Warehouse instance permission to use your keys, you supply the Key Protect information on the Console Administration -> Settings -> Manage Keys tab. You must provide the name of the Key Protect instance and the key that was created in the "Creating or adding a key in Key Protect" section. After the information is provided, you then migrate the instance to Key Protect managed keys.

Access Control

Audit and Privileges view access should be secured. Security administrative access should only be given to authorized users. Audit policy defined and in use views should never be available to the PUBLIC. Privileges view should not be available to PUBLIC. The service will reinforce this with every update.

There may be certain use cases where you will need to deviate from the secured configuration. If you wish to keep this as a permanent setting, please contact IBM Support.

Upgrade

Upgrading your system to Db2 Warehouse SaaS Next Generation

In order to upgrade existing systems to the next generation (Gen 3) of IBM® Db2® Warehouse as a Service, a self-service upgrade tool has been provided and IBM Db2 Warehouse SaaS users on IBM Cloud Cloud will be invited individually to start using the tool. Upgrading to the next generation of IBM Db2 Warehouse SaaS, unlocks opportunities to use IBM Cloud Object Storage (COS) to store database data while increasing performance of your system and saving on storage costs. It also unlocks the use of open data formats such as DATALAKE tables, allowing for seamless access to other data within your enterprise for integrated workloads.

Accessing the Upgrade Tool

1. Log in to IBM Cloud and navigate to the IBM Db2 Warehouse SaaS instance you wish to upgrade by going to Resource List > Databases > Service Details > Manage.
2. Locate the “System upgrade available: Upgrade to the newest version of IBM Db2 Warehouse SaaS”, then click Learn More to begin the upgrade process.

Step 1: Create a New Instance

A duplicate of the current (Gen 2) IBM Cloud Classic IBM Db2 Warehouse SaaS instance will be provisioned on the new Gen 3 infrastructure, with an estimated provisioning time displayed. All instance specifications, including compute and storage resources, will be seamlessly transitioned from the IBM Cloud Classic infrastructure to the Gen 3 environment. Once provisioning is complete, the new instance name will appear, and you can launch into the instance. At this stage, the instance will not contain any data until Step 2 is completed.

Step 2: Restore and Test Your New System (optional step)

You can choose an existing backup or create a new backup from the prior instance to restore onto the newly provisioned instance, bringing in all data. During the restore process, the estimated restore time will be displayed. At this time, you can point your applications to the new server instance to verify that your existing applications and workloads are functioning as expected. This step allows you to perform validation checks before triggering the final upgrade (Step 3). All identity and access management configurations will also be transferred. This step can be repeated as needed to sync data from the old instance to the new one.

Step 3: Final Upgrade

This step completes the upgrade process. A full backup of the database from the prior system will be taken and restored onto the new Gen 3 system. Both the hostname from the previous system and the new hostname for the Gen 3 instance will remain active, ensuring that applications, including those inserting and querying data, are routed to the upgraded Gen 3 environment. However, any private connections will need to be explicitly updated to point to the new instance, as described in more detail in the FAQ section. The prior system will still exist but will no longer be accessible for connections, effectively rendering it offline. Billing for both systems will continue unless the old system is deleted. At this stage, billing for the new system will begin.

Step 4: Delete Your Old System

Once the final upgrade in Step 3 is completed and the new instance is in use, you can delete the old IBM Db2 Warehouse SaaS instance. To do this, go to the actions dropdown on the manage page. Deleting the old system ensures that you avoid unnecessary charges and maintenance of outdated resources.

Step 5: Update Your Software

After the previous self-service steps were performed, your system will be at the same level of software as your previous system. The final step is to update the software and that can be achieved by either opening an IBM Support Ticket or clicking the Update Now button in your console.

For information about posting questions on a forum or opening a support ticket, see [Help & support](#).

Upgrade FAQs

What do I need to do with my applications before upgrade?

For users of the Db2 Warehouse on Cloud REST API

Ensure your application is using v4 of the IBM Db2 Warehouse SaaS on Cloud REST API. Additionally, all API requests must include the `x-deployment-id` header, or an error will be returned when using the next-generation warehouse. More information about the REST API can be found [here](#).

For users of the Db2 runtime client

Please ensure you are using Db2 version 11 clients or later. If you are using a connectivity driver supplied by a third party, verify with the vendor that it is compatible with Db2 version 11. Also, ensure that all connections use SSL via port `50001`, as non-SSL connections are not supported in the next-generation warehouse.

CLPPlus IDA functionality

The IDA functionality of CLPPlus (which allows development and registration of UDX in C++, R, Python) is no longer supported for use with IBM Db2 Warehouse SaaS.

For users of IBM Lift

Lift users will need to download the latest version of the Lift tool from IBM. This will be required after the tool is updated to use the v4 DBAPI.

What items will IBM retain for me, and what will I need to recreate?

Database-level objects

All your database-level objects, such as tables, views, statistics, and procedures, will remain intact after the upgrade.

User Credentials

All user credentials will be upgraded.

UDXes (User-Defined Extensions)

UDXes written in C++ will be upgraded successfully. However, new UDXes *cannot be registered* after the upgrade.

Server Connections

- Incoming connections to your Gen 2 instance (created via `CREATE SERVER`) may need to be reestablished.
- Outgoing connections from your Gen 2 instance (created via `CREATE SERVER`) will be upgraded. For private/vpn, you need to ensure the two machines can communicate.

Landing Zone Files

Files stored in your Gen 2 landing zone will *not be persisted* during the upgrade. Be sure to download and back up any files prior to upgrade, as IBM will not be responsible for lost contents in the landing zone.

Auditing

Your audit policies and data will be retained. During the upgrade, auditing on your v2 instance will be halted and started on the v3 instance with the same policies. Any audit data stored outside the AUDIT tables will be copied asynchronously. If you are auditing to tables, support for this configuration will end, and it's recommended to reconfigure your audit data to be written to an S3 cloud object storage (COS) instance. If you're already auditing to COS, your configuration will be upgraded, but you may notice a change in audit report format. For more details, see our [documentation](#). To reconfigure to COS, follow [these steps](#). If unsure of your current configuration, go to Audit under Administration, where you'll be prompted if you're using audit to tables.

Key Protect

Your Key Protect Integration configuration will be retained.

What is the testing period after the new system is created?

After successfully creating the new system (Step 1), you have a 14-day period to test and complete the final upgrade process (Step 3). The trial end date will be displayed, and once the 14-day period is over, billing for the new instance will automatically begin. If the prior system has not been deleted by this point, it will also continue to incur charges.

What should I do if I encounter an error during any of the steps?

While the upgrade process is robust, in the event an error does happen, you will receive a prompt stating, *"There was an error during this process. Please open a support ticket."* In such cases, simply open a support ticket as usual, and our support team will assist you in resolving the issue promptly.

What should I expect after the upgrade?

The self-service upgrade tool, will upgrade your system to the new generation of IBM Db2 Warehouse SaaS and VPC Gen2 infrastructure on IBM Cloud. It will retain all other levels of software, including the underlying database. In order to take advantage of the capabilities in the new generation of the product, a database update option will be available to be applied.

For most customers, you can resume working with your data as before. However, certain customers may encounter specific circumstances requiring further actions:

Private Connections from IBM Cloud VPC: If you have workloads running on IBM Cloud VPC, you can leverage private connectivity (Private Link) to connect securely to your Db2 Warehouse instances using IBM Cloud's private network. Follow the steps outlined [here](#).

To enable private connectivity:

- First, enable private connectivity through the console.
- Then, create a *Virtual Private Endpoint Gateway* on your VPC. The private endpoint and port will be provided in the Connections section of the console.



Note: Private connectivity is a *regional service*, meaning that the Virtual Private Endpoint Gateway must be created in the same region as your Db2 Warehouse on Cloud instance.

VPN Connectivity: If you currently connect via VPN, follow the instructions in the [VPN Connectivity guide](#) to ensure proper setup after the upgrade.

Using the Db2 database assistant

Chat with the IBM Db2 database assistant to help you complete tasks, find out about your system status and statistics, or ask general questions about Db2.

Open the database assistant from the Db2 web console, and then ask it to help you with a range of Db2 tasks, including:

- Answering your questions about Db2
- Detecting and helping to resolve lock contention issues
- Monitoring your Db2 instance, including getting information about:
 - Active connections
 - SQL activity
 - Resource usage
 - Schemas, table spaces, indexes, and tables in the system
 - Users
 - Backups
 - Compute scaling
 - Storage scaling

The database assistant uses built-in skills and is trained on the official IBM Db2 documentation so that it can provide you with the most accurate responses.

Starting the database assistant

To get started with the database assistant:

1. Open your Db2 web console.
2. Click the database assistant chat icon.



3. Type your questions and requests in the chat.

What can the database assistant do?

You can ask the database assistant to complete the actions listed in the following table.



Note: Depending on the database workload, the database assistant might take some time to respond to certain prompts, such as requesting a database summary or getting details about the largest tables by storage size.

Category	Action	Description	Example prompt phrases
Product information	Answer your questions about Db2	Ask the assistant anything about Db2 and it will generate an answer based on its knowledge of the product. The assistant is trained on the official IBM Db2 documentation	- What's new in Db2? - Tell me about the MON_GET_PKG_CACHE_STMT table function - How do I troubleshoot Db2? - What is error code SQL0911N? - How can I manage my database workload?
Instance information	Show backups	Get information about available backups in the system	- Get the list of backups - What backups are available? - Show me my backups
	Show database summary	Get a summary of basic database details such as name, version, size, number of tables, number of schemas, uptime, and more	- Report on my database - Check my database details - Tell me about this instance

	Show database version	Get the version of Db2	<ul style="list-style-type: none"> - What Db2 version am I running? - What is the database version? - Current Db2 database version - Show me the database version
	Show scaling resources	Get information about how system resources are configured for scaling	<ul style="list-style-type: none"> - Show me the system scaling settings - What is the scaling configuration? - Display compute scaling settings
	Show system settings	Get a summary of the host names and endpoints configured in your system	<ul style="list-style-type: none"> - Show my system settings - Display database endpoint settings - Describe db connection details
	Show users	Get a list of all users from the console	<ul style="list-style-type: none"> - Display users in the console. - Get me a list of users - Who is using the console?
	Open a specific console page	Navigate directly to a specific page within the Db2 web console	<ul style="list-style-type: none"> - Take me to the monitoring profile settings - Open my authorization settings - Show me the locking event monitor page - Take me to the SQL editor page
Workload information	Show active connections	Get a list of all current connections to the system	<ul style="list-style-type: none"> - Show me all connections - Display inflight connections - List the current applications
	Show running queries	Get a list of all currently running SQL queries and in-flight executions in the system	<ul style="list-style-type: none"> - Show me current SQL queries - Get all current queries - Show me in-progress SQL
	Show response times	Get a summary of query response times in the system	<ul style="list-style-type: none"> - Show me query response times - Query response times from 1pm to 4pm - Query response time summary
	Show query throughput	Get information about query processing rates and throughput	<ul style="list-style-type: none"> - Get query throughput - What is the query processing rate from the last 2 hours? - How many rows have been read in the last hour?
Storage information	Show table spaces	Get a list of table spaces in the system and their storage utilization	<ul style="list-style-type: none"> - Obtain the table spaces list - List all tablespace names - Show table spaces
	Show total storage	Get storage details about the system	<ul style="list-style-type: none"> - Display my storage utilization statistics - How much disk storage capacity have I used? - Report on my storage consumption
Resource utilization	Show CPU usage summary	Get a summary of how much CPU the system is using	<ul style="list-style-type: none"> - Show me CPU usage for the past 3 days - What was the system CPU load between August 14 and 15 2024? - Analyze processor utilization from the last 24 hours

	Show IO usage summary	Get a summary of disk usage over a specified period	<ul style="list-style-type: none"> - Analyze IO usage from the last 48 hours - Display IO utilization for the last 7 days - Show me I/O usage
	Show memory usage summary	Get a summary of how much memory was used by the system over a specified period	<ul style="list-style-type: none"> - Analyze memory usage for the last 15 days - Get a summary of memory details - Show me memory usage from 6 September 2024
	Show database performance	Get information about how your database is being used, including performance statistics	<ul style="list-style-type: none"> - Get database performance from 8 Jan 2025 to 10 Jan 2025 - How is my database performing today? - Show me database performance stats for the past week
Database objects	Show indexes for a specified schema	Get a list of indexes for a given schema	<ul style="list-style-type: none"> - Get index details - Show indexes for the schema schema-name-1 - Fetch indexes
	Show largest tables by rows	Get a ranked list of the largest tables in the system based on row count	<ul style="list-style-type: none"> - Get the largest 10 tables - Display the 3 largest tables by row count - What is my largest table?
	Show largest tables by rows for a specified schema	Get a ranked list of the largest tables in a given schema	<ul style="list-style-type: none"> - What are the top 3 tables for the schema "schema-name"? - Get the largest tables by row count from the schemas "schema-name-1" and "schema-name-2" - Show me the 3 biggest tables by rows from schema-name-1, schema-name-2 schemas
	Show largest tables by storage	Get a ranked list of the largest tables in the system based on storage size	<ul style="list-style-type: none"> - Rank the top 10 tables by storage size - Identify my largest tables by storage - Which table uses the largest amount of storage?
	Show largest tables by storage for a specified schema	Get a ranked list of the largest tables in a given schema based on storage size	<ul style="list-style-type: none"> - Get the top 4 tables by storage for the schema schema-name-1 - What is the largest table by storage size in the schema schema-name-1? - Show the top 10 largest tables by storage from the schema-name-1 schema
	Show schemas	Get a list of schemas in the system	<ul style="list-style-type: none"> - Display all schemas - Provide a list of schemas - Fetch me a list of schemas in the database
	Show tables	Get a list of tables in the system	<ul style="list-style-type: none"> - View the tables in the schema "schema-name-1" - Get a tables list from the schema-name-1, schema-name-2, and schema-name-3 schemas - List my tables

	Show views	Get a list of views in the system	<ul style="list-style-type: none"> - List my views - Display the views for the schema-name-1 schema - Collect all views
	Show outdated statistics	Get information about tables or schemas that have outdated statistics	<ul style="list-style-type: none"> - Get a list of list of tables in the schema schema-name-1 that have outdated stats - Get the outdated stats for the schema schema-name-1 with at least 30% row changes - Give me list of tables that have outdated stats in the last 2 hours
	Get updated statistics	Get information about tables or schemas that have updated statistics	<ul style="list-style-type: none"> - Get updated statistics - Show me updated stats for the schema schema-name-1 with at least 30% row changes - Get updated stats for the schema schema-name-1 with row changes of 20% or more in the last 5 hours
Troubleshooting	Show most active connections	Get details about the most active connections	<ul style="list-style-type: none"> - What are the top 5 active connections - Display the top 8 connections in the database - Get the most active connections
	Show how time is spent	Get details about how much time was spent by the system on various tasks	<ul style="list-style-type: none"> - List wait times and bottlenecks - How is time being spent in my database? - Get time spent details
	Show lock waits	Get a list of blocking and waiting connections in the system	<ul style="list-style-type: none"> - Troubleshoot lock waits and deadlocks - Report all lock waits - Show me lock waits from the last two hours
	Show longest running queries	Get information about the top queries in the system ranked by how long they have been running	<ul style="list-style-type: none"> - Display the longest running SQL queries - What are the top 5 queries? - Show long running SQL statements

Db2 database assistant skills

Troubleshooting the Db2 database assistant

If you find that you are not getting the answers and results that you expect from the Db2 database assistant, try one or more of the following troubleshooting strategies:

- Rephrase the question. The assistant might not understand your phrasing. Try using different wording, if possible.
- Restart the conversation. The assistant might be getting confused by other things you have discussed with it.
- Ensure that you are asking about Db2. If you are asking about something else, it might be outside of the assistant's capabilities.
- Try asking about something specific. If your question is too broad, the assistant is more likely to return results that aren't relevant to your request.
- Ensure that you are not filtering by version. If an answer to your question isn't available in the current documentation release, the assistant won't find it. If you're filtering by version, some content might be missing. To broaden the search, type "change version" and then choose "Any".

Data privacy and opting out

When you interact with the Db2 database assistant, your questions and requests are processed by IBM watsonx Orchestrate, which is outside of your Db2 instance. Only the text that you input into the assistant gets processed by watsonx Orchestrate. None of your actual data or metrics leave your Db2 instance at any time.

To opt out of the Db2 database assistant, contact IBM Support.

Communities

There are user-driven communities that you can join for information, tutorials, discussions, and help from professional Db2 users. And they're all free to join!

- [International Db2 Users Group \(IDUG\)](#) IDUG® is an independent, not-for-profit, user-run organization whose mission is to support and strengthen the information services community. It provides the highest-quality education and services that are designed to promote the effective use of Db2.
- [Db2 Community on IBM Communities](#) A Db2 developer community.
- [Stack Overflow](#) A support forum and community for developers.

FAQs

This is a collection of frequently asked questions (FAQ) about the IBM® Db2® Warehouse as a Service service.

How do I sign up for Db2 Warehouse on Cloud?

You can provision an instance of IBM Db2 Warehouse SaaS directly through the IBM Cloud® catalog. You can [create a free IBM Cloud account](#) and get an IBM Cloud credit of \$200 that you can use towards IBM Db2 Warehouse SaaS.

How do I choose the Db2 Warehouse on Cloud plan that's right for me?

IBM Db2 Warehouse SaaS offers several elastic data warehouse configurations to meet your workload requirements. For more information, see [About](#).

How do I generate credentials for my instance?

1. Log into [IBM Cloud](#).
2. Open your [Resource list](#).
3. Under **Databases**, locate your IBM Db2 Warehouse SaaS instance and click on the service name.
4. Select the **Service credentials tab > New credentials+ > Add** in order to generate your service admin credentials.
5. Expand **View credentials**, which displays your service connectivity information including your admin credentials (username and password).
6. The admin credentials can be used to connect to both Db2 and the web console.

Now that I've generated credentials, how do I access my Db2 Warehouse on Cloud instance?

You can access your IBM Db2 Warehouse SaaS instance through several methods, including a dedicated web console and a REST API. For more information, see [Interfaces](#).

What's managed for me with Db2 Warehouse on Cloud?

IBM handles all of the software upgrades, operating system updates, and hardware maintenance for your IBM Db2 Warehouse SaaS instance. IBM also preconfigures Db2 parameters for optimal performance across analytical workloads, and takes care of encryption and regular backups of your data.

The service includes 24x7 health monitoring of the database and infrastructure. In the event of a hardware or software failure, the service is automatically restarted. Because IBM Db2 Warehouse SaaS is a fully-managed SaaS offering, you do not get SSH access or root access to the underlying server hardware, and cannot install additional software.

Where can I find more information about Db2 Warehouse on Cloud?

In addition to the IBM Cloud documentation site, there is a wide range of information about the underlying Db2 engine functionality in the [Knowledge Center](#). Updates to the service are posted on our [What's New](#) page.

You can find pricing information and deploy a IBM Db2 Warehouse SaaS instance through the IBM Cloud [catalog](#) page for IBM Cloud. To learn more, contact [IBM Sales](#).

Where can I find help for a problem that I'm having?

For information about posting questions on a forum or opening a support ticket, see [Help & support](#).

Help and support

Here you'll find answers to common questions about using IBM® Db2® Warehouse as a Service.

Db2 Warehouse on Cloud service is not available

The IBM Db2 Warehouse SaaS service in IBM Cloud might not be available.

What's happening

An attempt to create a data warehouse fails with a 500 error, or an SDP load reports an SQL error.

Why it's happening

The IBM Db2 Warehouse SaaS service in IBM Cloud is undergoing a scheduled maintenance procedure or a known outage occurred.

How to fix it

You can determine the status of the IBM Db2 Warehouse SaaS service in IBM Cloud by checking the [IBM Cloud: Status](#) page.

Getting help and support for Db2 Warehouse on Cloud

If you have problems or questions when using IBM Db2 Warehouse SaaS, you can get help by searching for information or by asking questions through a forum. You can also open a support ticket.

When using the forums to ask a question, tag your question so that it is seen by the IBM Cloud® development teams.

- If you have technical questions about developing or deploying an app with IBM Db2 Warehouse SaaS, post your question on [Stack Overflow](#) and tag your question with "bluemix" and "db2".
- For questions about the service and getting started instructions, use the [IBM developerWorks® dW Answers](#) forum. Include the "bluemix" and "db2" tags.

See [Getting help](#) for more details about using the forums.

For information about opening an IBM support ticket, or about support levels and ticket severities, see: [Working with support cases](#).

© Copyright IBM Corporation 2025

IBM Corporation
New Orchard Road
Armonk, NY 10504

Produced in the United States of America
2025-03-11

IBM, the IBM logo, and [ibm.com](https://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at <https://www.ibm.com/legal/copytrade>.

This document is current as of the initial date of publication and may be changed by IBM at any time. Not all offerings are available in every country in which IBM operates.

THE INFORMATION IN THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT ANY WARRANTY, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

IBM products are warranted according to the terms and conditions of the agreements under which they are provided.

