ALIBABA CLOUD

Alibaba Cloud

Data Transmission Service
Product Introduction

Document Version: 20220712

Alibaba Cloud

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.

2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.

3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.

4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).

5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.

6. Please directly contact Alibaba Cloud for any errors of this document.

Document conventions

Style	Description	Example
Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	**Danger:** Resetting will result in the loss of user configuration data.
Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	**Warning:** Restarting will cause business interruption. About 10 minutes are required to restart an instance.
Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	**Notice:** If the weight is set to 0, the server no longer receives new requests.
Note	A note indicates supplemental instructions, best practices, tips, and other content.	**Note:** You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click **Settings**> **Network**> **Set network type**.
Bold	Bold formatting is used for buttons , menus, page names, and other UI elements.	Click **OK**.
`Courier font`	Courier font is used for commands	Run the `cd /d C:/window` command to enter the Windows system folder.
Italic	Italic formatting is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	This format is used for an optional value, where only one item can be selected.	`ipconfig [-all|-t]`
{} or {a|b}	This format is used for a required value, where only one item can be selected.	`switch {active|stand}`

Table of Contents

1.What is DTS?

Alibaba Cloud Data Transmission Service (DTS) is a real-time data streaming service. DTS supports data transmission between data sources such as relational, NoSQL, and online analytical processing (OLAP) databases. DTS provides the data synchronization, data migration, change tracking, data integration, and data processing features. This enables you to manage data within a secure, scalable, and high-availability architecture.

Benefits

Compared with traditional data migration and synchronization tools, DTS has the following advantages: high compatibility, high performance, security, reliability, and ease of use. This allows you to simplify data transmission and focus on your application development.

- **Ease of use**

  The DTS console is a visual management interface that provides a wizard-like process for creating DTS tasks.

- **Diverse features**
  - DTS allows you to migrate or synchronize data between heterogeneous databases that run on different database engines and have different architectures.
  - DTS provides multiple data replication modes, including data migration, data synchronization, data integration, and change tracking.

- **High performance**
  - DTS uses servers with high specifications to ensure the performance of each data synchronization or migration instance.
  - DTS has its core infrastructure optimized and can achieve a peak data transmission rate of 70 MB/s.
  - DTS supports concurrent compressed data transmission that minimizes the bandwidth utilization.

- **High security and reliability**
  - DTS supports disaster recovery of nodes. A failing task connection can be recovered within seconds.
  - DTS supports resumable transmission to resolve transmission interruptions caused by exceptions such as hardware and network failures.
  - DTS allows you to manage access to DTS tasks with fine-grained custom policies created by using Resource Access Management (RAM).
  - DTS supports data verification to ensure data integrity and consistency between the source and destination databases.

For more information about DTS, see Benefits and Scenarios.

Features

Feature	Description	References

Feature	Description	References
Data synchronization	You can use DTS to synchronize data between data sources in real time. This feature is suitable for the following scenarios: active geo-redundancy, geo-disaster recovery, zone-disaster recovery, cross-border data synchronization, cloud-based business intelligence (BI) systems, and real-time data warehousing.	• Overview of data synchronization scenarios • Data synchronization mode
Data migration	You can use DTS to migrate data between homogeneous and heterogeneous data sources. This feature is suitable for the following scenarios: data migration to Alibaba Cloud, data migration between databases within Alibaba Cloud, and database splitting and scale-out. **Note** Data migration can partly achieve data synchronization in specific scenarios. However, data migration and data synchronization differ in applicable scenarios, functionality, and billing. For more information about the differences between data migration and data synchronization, see the "What are the differences between data migration and data synchronization?" section of the FAQ topic.	• Overview of data migration scenarios • Data migration mode
Data integration	Data integration is an advanced feature on top of data migration. You can configure a scheduling policy to migrate schemas and historical data from a source database to a destination database on a regular basis. The data integration feature allows you to flexibly build a data warehouse such as an offline data warehouse.	Data integration mode
Change tracking	You can use DTS to track incremental data from self-managed MySQL databases, ApsaraDB RDS for MySQL instances, PolarDB for MySQL clusters, PolarDB-X instances, and self-managed Oracle databases in real time. Then, you can consume the tracked data as needed. This feature is suitable for the following scenarios: cache updates, business decoupling and asynchronous data processing, real-time data synchronization between heterogeneous databases, and real-time data synchronization that involves extract, transform, load (ETL) operations.	Change tracking mode
Data processing	You can use DTS to extract, transform, process, and load streaming data. This feature helps enterprises navigate real-time data processing and computing scenarios and empower enterprises to go digital.	What is ETL?What is ETL?

Supported databases

DTS supports a variety of data sources. For more information, see Supported sources and targets.

Management tools

You can use the following methods to create and configure DTS tasks and perform task O&M:

- DTS console: a web service page used to efficiently manage DTS tasks.
- API: provides API operations that you can call to perform all operations that are available in the DTS console.
- SDK: provides SDKs for a variety of programming languages to perform all operations that are available in the DTS console.

Pricing

For more information, see Billing methods.

Related Alibaba Cloud services

- Elastic Compute Service (ECS): Self-managed databases hosted on ECS instances can serve as the data sources or destinations for data migration, data synchronization, and change tracking tasks.
- ApsaraDB RDS: ApsaraDB RDS is a stable, reliable, and scalable online database service. ApsaraDB RDS provides a variety of solutions for disaster recovery, backup, restoration, and migration to facilitate database O&M.
- PolarDB: PolarDB is a next-generation relational database service that is developed by Alibaba Cloud. PolarDB is compatible with the MySQL, PostgreSQL, and Oracle database engines. PolarDB provides superior performance in storage and computing to meet the requirements of enterprises. Each PolarDB cluster provides a storage capacity of up to 100 TB and supports up to 16 nodes.
- PolarDB-X: PolarDB-X is a cloud-based distributed database service that is designed to handle the bottlenecks of standalone databases. PolarDB-X is developed in-house by Alibaba Cloud. PolarDB-X is highly compatible with MySQL protocols and syntax and supports automatic sharding, online smooth scaling, elastic scaling, and read/write splitting. PolarDB-X provides O&M capabilities throughout the lifecycle of databases.
- ApsaraDB for Redis: ApsaraDB for Redis is an online key-value storage database service that provides high performance and is compatible with open source Redis protocols.
- ApsaraDB for MongoDB: ApsaraDB for MongoDB is a MongoDB-compatible database service that is developed based on the Apsara distributed operating system and a high-reliability storage engine. ApsaraDB for MongoDB uses a multi-node architecture to ensure high availability, and supports elastic scaling, disaster recovery, backup and restoration, and performance optimization.

2.Supported databases

Data Transmission Service (DTS) supports data transmission between various data sources, such as relational database management systems (RDBMS), NoSQL databases, and online analytical processing (OLAP) databases. This topic describes the database types, migration types, synchronization types, synchronization topologies, and data change types that are supported by DTS.

> **Note** For more information about terms related to migration types, synchronization types, and change tracking, see Terms.

Data synchronization

You can use DTS to synchronize data between data sources in real time. This feature is suitable for the following scenarios: active geo-redundancy, geo-disaster recovery, zone-disaster recovery, cross-border data synchronization, query load balancing, cloud BI systems, and real-time data warehousing.

For more information about the supported database types, database engine versions, synchronization types, and how to configure a data synchronization task, see the following tables or Overview of data synchronization scenarios.

- Synchronize data from a self-managed MySQL database or an ApsaraDB RDS for MySQL instance
- Synchronize data from a PolarDB for MySQL cluster
- Synchronize data from a PolarDB-X instance
- Synchronize data from a PolarDB for Oracle cluster
- Synchronize data from a self-managed Oracle database
- Synchronize data from a self-managed PostgreSQL database or an ApsaraDB RDS for PostgreSQL instance
- Synchronize data from a self-managed SQL Server database or an ApsaraDB RDS for SQL Server instance
- Synchronize data from a self-managed MongoDB database or an ApsaraDB for MongoDB instance

  > Note
- Synchronize data from a self-managed Redis database or an ApsaraDB for Redis instance

  > Note
- Synchronize data from a TiDB database

Note

- Synchronize data from a Db2 for LUW database
- Synchronize data from a Db2 for i database
- Synchronize data from a DMS logical database

Note

Data migration

You can use DTS to migrate data between homogeneous and heterogeneous data sources. Typical scenarios include data migration to Alibaba Cloud, data migration between instances within Alibaba Cloud, and database splitting and scale-out.

For more information about the supported databases, database engine versions, migration types, and how to configure a data migration task, see the following tables or Overview of data migration scenarios.

- Migrate data from a self-managed MySQL database or an ApsaraDB RDS for MySQL instance
- Migrate data from an ApsaraDB RDS for MariaDB TX instance

Note

- Migrate data from an ApsaraDB RDS for PPAS instance

Note

- Migrate data from a PolarDB for MySQL cluster
- Migrate data from a PolarDB for Oracle cluster
- Migrate data from a PolarDB-X instance
- Migrate data from a PolarDB for PostgreSQL cluster
- Migrate data from a MaxCompute project

Note

- Migrate data from a self-managed Oracle database
- Migrate data from a self-managed PostgreSQL database or an ApsaraDB RDS for

PostgreSQL instance

- Migrate data from a self-managed SQL Server database or an ApsaraDB RDS for SQL Server instance

  Note

- Migrate data from a self-managed MongeDB database or an ApsaraDB for MongoDB instance
- Migrate data from a self-managed Redis database or an ApsaraDB for Redis instance

  Note

- Migrate data from a TiDB database

  Note

- Migrate data from a Db2 for LUW database
- Migrate data from a Db2 for i database
- Migrate data from a Teradata database

  Note

- Migrate data from a self-managed HBase database
- Migrate data from a database hosted on a third-party cloud

Note

- Data migration can partly achieve data synchronization in specific scenarios. However, data migration and data synchronization differ in scenarios, supported databases, features, and billing. For more information about the differences between data migration and data synchronization, see the "What are the differences between data migration and data synchronization?" section of the FAQ topic.
- If you need to migrate incremental data for a long period of time, we recommend that you use data synchronization, which can achieve better performance and network stability.

Change tracking

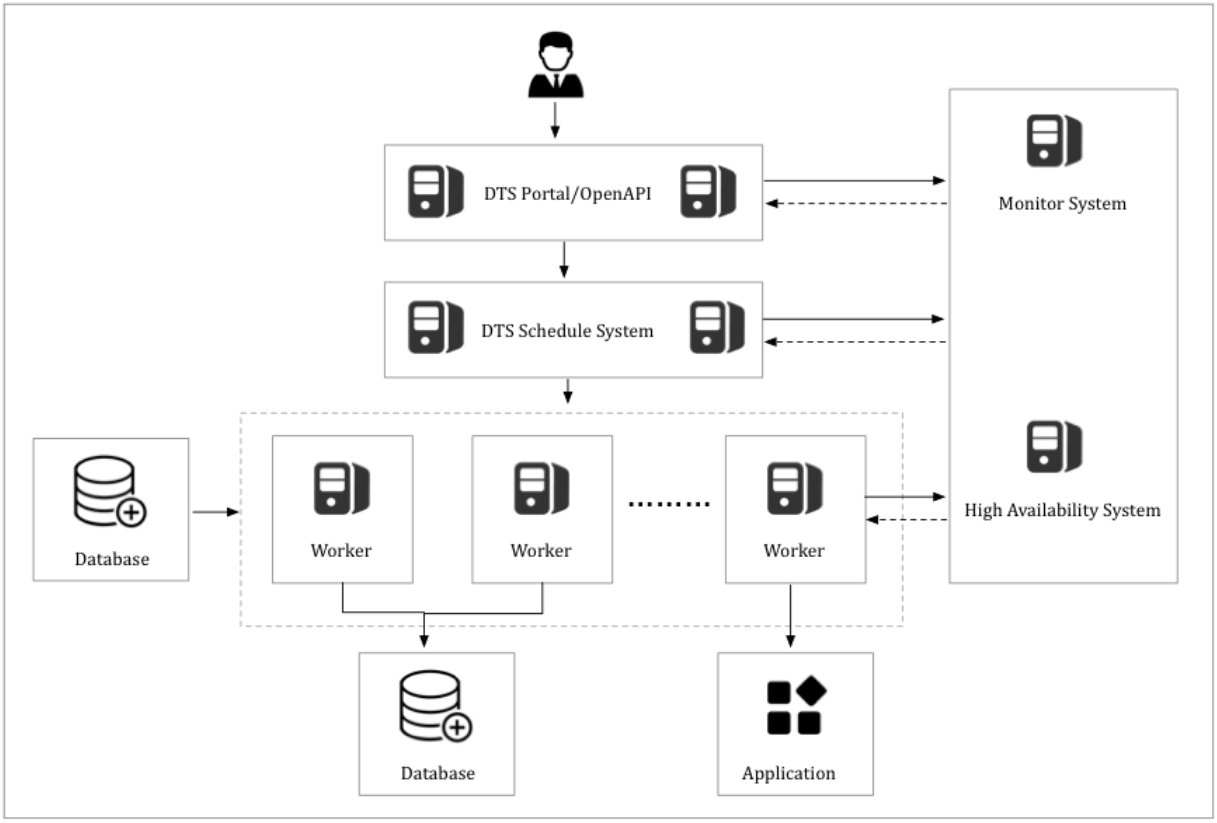
You can use DTS to track data changes from databases in real time. Then, you can consume the tracked data in the following scenarios: cache updates, asynchronous business decoupling, data synchronization between heterogeneous data sources, and data synchronization with extract,

transform, and load (ETL) operations. Change tracking allows you to track incremental data of a variety of databases in real-time, such as a self-managed MySQL database and an ApsaraDB RDS for MySQL instance. You can use different clients to consume tracked data, such as SDK clients and Flink clients. For more information about the supported change tracking solutions and how to configure a change tracking task, see the following table or Change tracking scenarios.

3.System architecture and design concepts

This topic describes the system architecture of Data Transmission Service (DTS) and the basic design concepts of its main features.

System architecture



Architecture description

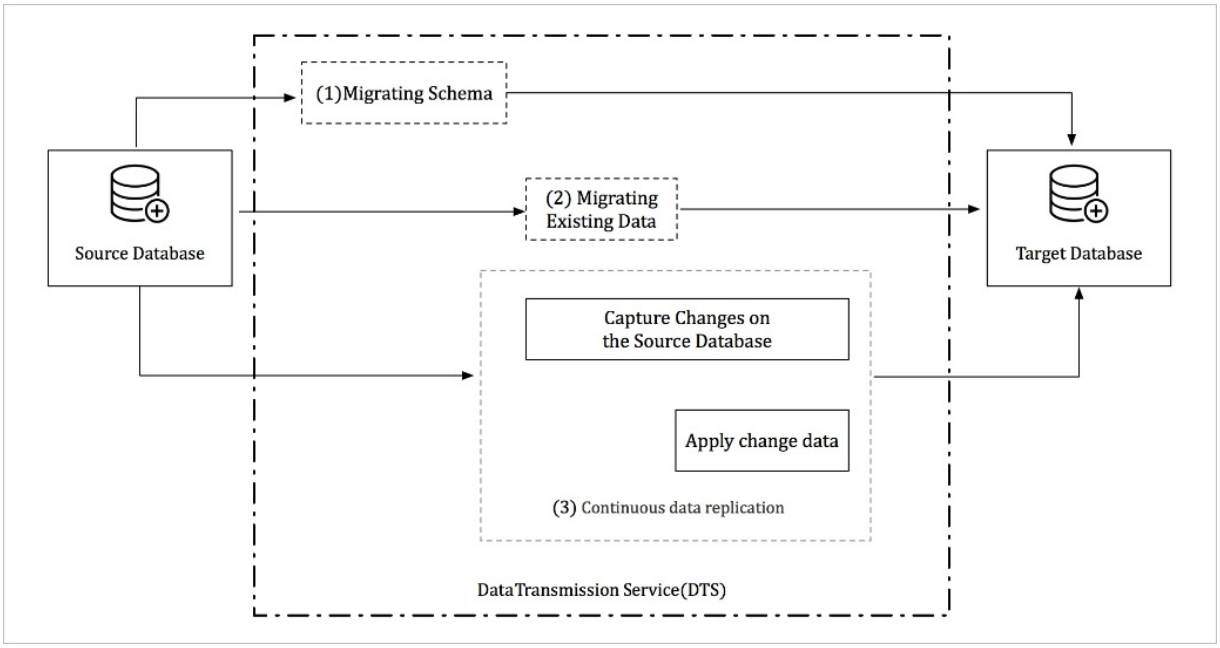
- High availability

  Each module in DTS has primary and secondary nodes to ensure high availability. The disaster recovery module runs a health check on each node in real time. When a node exception is detected, the module only requires a few seconds to switch the channel to another healthy node.

- Connection reliability

  To ensure the connection reliability of change tracking and data synchronization channels, the disaster recovery module checks for any changes, such as changes of data source address. When a change of data source address is detected, the module dynamically allocates a new connection method to ensure the stability of the channel.

Design concept of data migration



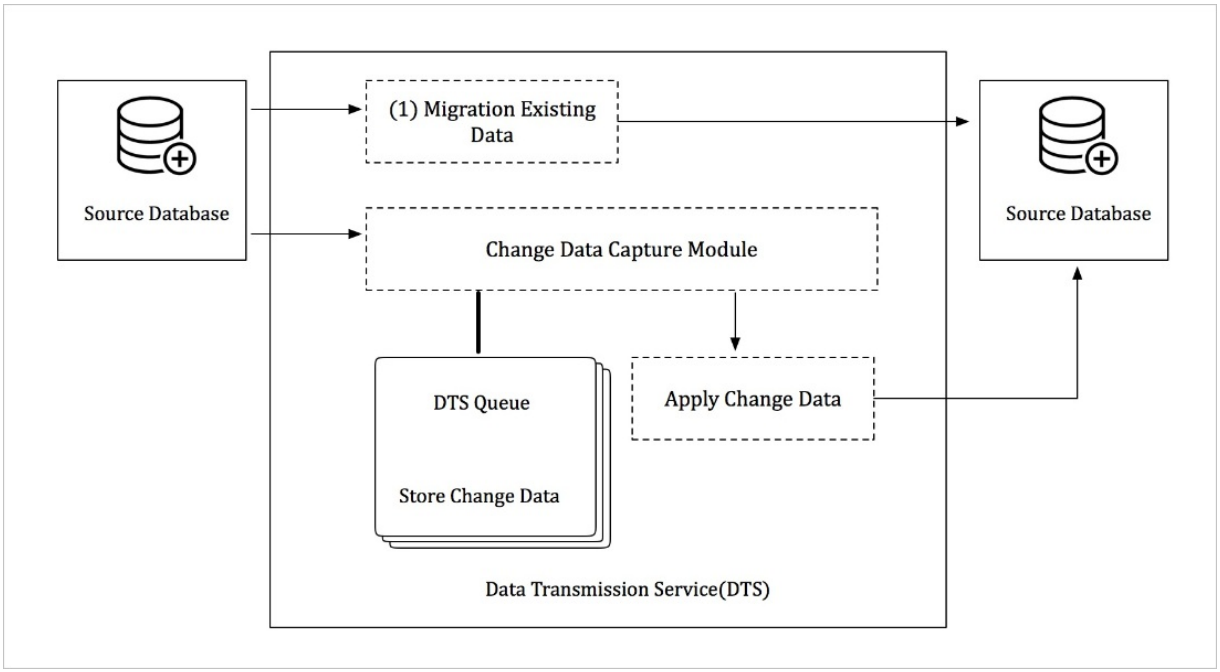
Data migration supports schema migration, full data migration, and incremental data migration. To migrate data with minimized downtime, you must select schema migration, full data migration, and incremental data migration.

For a migration between heterogeneous databases, DTS reads the source database schema, translates the schema into the syntax of the destination database, and imports the schema to the destination instance.

A full data migration requires a long period of time. In this process, incremental data is continuously written into the source instance. To ensure data consistency, DTS starts the incremental data reading module before full data migration. This module obtains the incremental data from the source instance and parses, encapsulates, and locally stores the data.

When a full data migration is complete, DTS starts the incremental data loading module. This module retrieves the incremental data from the incremental data reading module. After reverse parsing, filtering, and encapsulation, the incremental data is migrated to the destination instance.

Design concept of real-time data synchronization



The real-time data synchronization feature can be used to synchronize incremental data between any two data sources in real time. The real-time data synchronization from OLTP databases to OLAP databases is supported.

The process of creating a data synchronization channel is as follows:

- Initial synchronization

  DTS synchronizes the schemas and historical inventory data of the objects that you have selected from the source instance to the destination instance.

- Real-time synchronization of incremental data

  After initial synchronization is complete, DTS synchronizes incremental data from the source instance to the destination instance in real time.

DTS uses the following modules for real-time synchronization of incremental data:

- Incremental data reading module

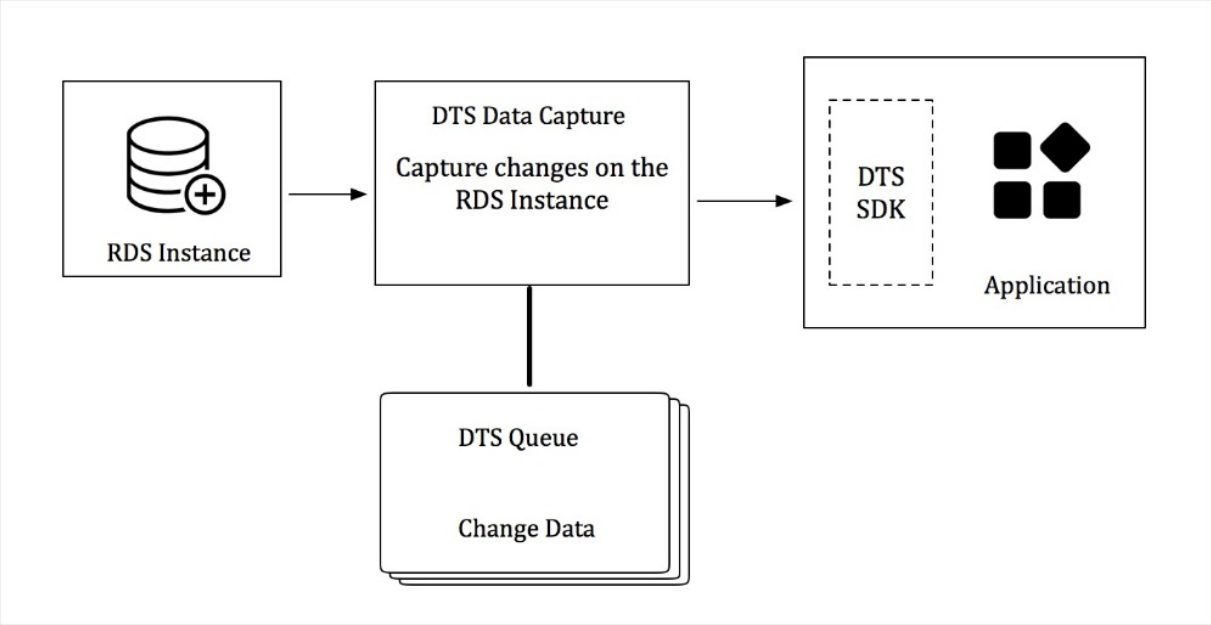
  The incremental data reading module obtains raw data from the source instance. After parsing, filtering, and syntax conversion, the data is locally stored. The incremental data reading module connects to the source instance by using a database protocol and obtains incremental data from the source instance. If the source database is ApsaraDB RDS for MySQL, the incremental data reading module connects to the source database by using the binary log dump protocol.

- Incremental data loading module

  The incremental data loading module requests incremental data from the incremental data reading module and filters data based on the specified objects to be synchronized. Then, the incremental data loading module synchronizes data to the destination instance without compromising transactional sequence and consistency.

DTS ensures high availability of the incremental data reading module and incremental data loading module. When a channel exception is detected, the disaster recovery module switches the channel to a healthy node. This ensures high availability of the synchronization channel.

Design concept of change tracking



The change tracking feature allows you to obtain incremental data from an RDS instance in real time. You can subscribe to the incremental data on the change tracking server by using DTS SDKs. You can also customize data consumption rules based on your business requirements.

The incremental data reading module on the server side of DTS obtains raw data from the source instance. After parsing, filtering, and syntax conversion, the incremental data is locally stored.

The incremental data reading module connects to the source instance by using a database protocol and obtains the incremental data from the source instance in real time. If the source database is ApsaraDB RDS for MySQL, the incremental data reading module connects to the source database by using the binary log dump protocol.

DTS ensures high availability of the incremental data reading module and consumption SDK processes.

- When an exception is detected in the incremental data reading module, the disaster recovery module restarts the incremental data reading module on a healthy service node. This ensures high availability of the incremental data reading module.
- DTS ensures high availability of consumption SDK processes on the server. If you start multiple consumption SDK processes for the same channel, the server pushes incremental data to only one process at a time. If an exception occurs on a process, the server pushes data to another healthy consumption process.

4.Benefits

Data Transmission Service (DTS) supports data transmission between different data sources, such as relational database management system (RDBMS), NoSQL, and online analytical processing (OLAP) databases. Multiple data transmission features are available in DTS, including data migration, real-time change tracking, and real-time data synchronization. Compared with third-party data streaming tools, DTS provides multiple types of instances with high performance, security, and reliability. In addition, it is simple to create and manage instances.

Features

DTS supports data migration between homogeneous and heterogeneous data sources. For example, you can migrate data between MySQL databases or from Oracle databases to PolarDB for Oracle clusters. For migration between heterogeneous data sources, DTS supports schema conversion. For example, you can convert a synonym in Oracle to a synonym in PolarDB for Oracle.

DTS supports multiple data transmission modes, including data migration, real-time change tracking, and real-time data synchronization.

Real-time data synchronization supports one-way and two-way synchronization between two data sources. This feature is ideal for the following scenarios: geo-disaster recovery, active geo-redundancy, nearby application access, query load balancing, and real-time data warehousing.

DTS supports data migration with minimized downtime to ensure your application availability. The application downtime during data migration is reduced to minutes.

High performance

DTS uses servers with high specifications to ensure the performance of each data synchronization or migration instance.

The infrastructure of DTS has been optimized to ensure high-speed and reliable data migration. The peak rate of full data migration can reach 70 MB per second or 200,000 transactions per second (TPS).

Compared with traditional data synchronization tools, DTS provides better synchronization performance. You can use DTS to concurrently synchronize transactions and the incremental data of a single table. During peak hours, the data synchronization performance can reach 30,000 records per second (RPS).

DTS supports concurrent compressed data transmission that minimizes the bandwidth utilization.

> **Note** The performance indicators are for reference only. The data migration and data synchronization performance is affected by various factors such as the performance of the source and destination databases, network latency, network bandwidth, and instance specifications.

High reliability

DTS is implemented based on clusters. If a node in a cluster is down or faulty, the control center moves all tasks from this node to another healthy node in the cluster within seconds.

DTS provides a 24 x 7 mechanism for validating data accuracy in some instances to discover and rectify inaccurate data. This helps ensure data integrity.

Secure transmission protocols and tokens are used for authentication across DTS modules to ensure reliable data transmission. DTS also supports resumable transmission.

Ease of use

The DTS console is a visual management interface that can guide you through creating instances in a simple manner.

The DTS console provides task information for you to manage your tasks, such as task state, progress, and performance.

DTS supports resumable transmission and regularly monitors task states. If DTS detects an error such as network failure or system exception, it automatically fixes the error and restarts the task. If the error persists, you must manually check and restart the task in the DTS console.

Replication modes

DTS provides multiple data replication modes, including data migration, data synchronization, and change tracking. You can choose an ideal replication mode for your actual scenario.

The data synchronization mode can be used to synchronize data between data sources. You have the option to choose one-way or two-way data synchronization.

> **Note** Two-way data synchronization is available only for scenarios between MySQL databases, PolarDB for MySQL clusters, or ApsaraDB for Redis Enhanced Edition (Tair) instances.

The data synchronization mode can be used to distribute workloads among nodes in real time. This delivers high availability and load balancing and can be used as real-time data warehousing.

Data migration with minimized downtime

The data migration mode can be used to migrate data with minimized downtime. Source databases can remain operational during data migration. Your service downtime during data migration is reduced to minutes.

5.Features

5.1. Data synchronization

You can use DTS to synchronize data between two data sources in real time. This feature applies to multiple scenarios, such as active geo-redundancy, geo-disaster recovery, zone-disaster recovery, cross-border data synchronization, query load balancing, cloud BI systems, and real-time data warehousing.

Examples

For more information, see Overview of data synchronization scenarios.

Databases, initial synchronization types, and synchronization topologies

For more information, see Overview of data synchronization scenarios.

Supported objects to be synchronized

Objects that can be synchronized include databases, tables, and columns. You can specify objects that you want to synchronize.

Advanced features

Feature	Description	Reference
Add and remove objects to be synchronized	You can add and remove objects to be synchronized during data synchronization.	• Add an object to a data synchronization task • Remove an object from a data synchronization task
View and analyze synchronization performance	Data synchronization provides diagrams for analyzing the synchronization latency, RPS, and traffic statistics. This allows you to view the performance trend of a synchronization channel.	View the connection status and performance of data synchronization
ETL features	• Supports object name mappings at database, table, and column levels. With this feature, you can migrate data between two databases, tables, or columns that have different names. • Supports data filtering. With this feature, you can use SQL conditions to filter the required data in a specific table.	• Rename an object to be synchronized • Use SQL conditions to filter data

Statuses of a synchronization task

A synchronization task is a basic unit of real-time data synchronization. To synchronize data between two instances, you must create a synchronization task in the DTS console.

The following table describes the statuses of a synchronization task during creation and running.

Task status	Description	Available operations
Prechecking	A precheck is being performed before the synchronization task is started.	• View configurations for synchronization. • Delete the synchronization task. • Replicate synchronization configurations.
Precheck Failed	The synchronization task has failed to pass the precheck.	• Perform the precheck. • View synchronization configurations. • Modify objects to be synchronized. • Modify the synchronization speed. • Delete the synchronization task. • Replicate synchronization configurations.
Not Started	The synchronization task has passed the precheck but is not started.	• Perform the precheck. • Start the synchronization task. • Modify objects to be synchronized. • Modify the synchronization speed. • Delete the synchronization task. • Replicate synchronization configurations.
Performing Initial Synchronization	The initial synchronization is in progress.	• View synchronization configurations. • Delete the synchronization task. • Replicate synchronization configurations.

Task status	Description	Available operations
Initial Synchronization Failed	Data migration has failed during initial synchronization.	• View synchronization configurations. • Modify objects to be synchronized. • Modify the synchronization speed. • Delete the synchronization task. • Replicate synchronization configurations.
Synchronizing	The task is synchronizing data.	• View synchronization configurations. • Modify objects to be synchronized. • Modify the synchronization speed. • Delete the synchronization task. • Replicate synchronization configurations.
Synchronization Failed	An error occurred during synchronization.	• View synchronization configurations. • Modify objects to be synchronized. • Modify the synchronization speed. • Delete the synchronization task. • Replicate synchronization configurations.

Task status	Description	Available operations
Paused	The synchronization task is paused.	• View synchronization configurations. • Modify objects to be synchronized. • Modify the synchronization speed. • Delete the synchronization task. • Replicate synchronization configurations.

If a synchronization task is in the **Synchronization Failed** or **Initial Synchronization Failed** state, the task will be started and resumed within seven days. To stop data synchronization, you must manually pause or stop the synchronization task.

References

- Synchronization topologies
- Data type mappings for schema synchronization
- Terms

5.2. Data migration

You can use DTS to migrate data between various types of data sources. Typical scenarios include data migration to Alibaba Cloud, data migration between instances within Alibaba Cloud, and database splitting and scale-out. DTS supports data migration between homogeneous and heterogeneous data sources. It also supports extract, transform, and load (ETL) features such as object name mapping and data filtering.

Database and migration types

For more information, see Overview of data migration scenarios.

Easy to use

After you configure the source instance, destination instance, and objects to be migrated, DTS automatically completes the entire data migration process and ensures service continuity.

Migration types

DTS supports **schema migration**, **full data migration**, and **incremental data migration**. You can select all of the supported migration types to ensure service continuity.

Migration types	Description

Migration types	Description
schema migration	DTS migrates the schemas of the required objects from the source database to the destination database. Tables, views, triggers, and stored procedures can be migrated. For schema migration between heterogeneous databases, DTS converts the schema syntax based on the syntax of the source and destination databases. For example, it converts the NUMBER data type in Oracle databases to the DECIMAL data type in MySQL databases.
full data migration	DTS migrates historical data of the required objects from the source database to the destination database. If you select only schema migration and full data migration, new data generated in the source database will not be migrated to the destination database. **Note** To ensure data consistency, do not write new data into the source database during full data migration. To migrate data with minimal downtime, you must select **schema migration**, **full data migration**, and **incremental data migration** when configuring a data migration task.
incremental data migration	DTS retrieves static snapshots from the source database and migrates the snapshot data to the destination database. Then, DTS synchronizes the incremental data generated in the source database to the destination database in real time. **Note** During incremental data migration, data between the source and destination databases are synchronized in real time. The migration task does not automatically stop. You must manually stop the migration task.

ETL features

Data migration supports the following ETL features:

- Object name mapping for columns, tables, and databases: you can migrate data between two columns, tables, or databases that have different names.
- Data filtering: you can use SQL conditions to filter the required data in a specific table. For example, you can specify a time range to migrate only the latest data.

Alerts

DTS supports sending alerts for data migration tasks. If an error occurs in a task, DTS immediately sends an SMS alert to the task creator. The alerts allow the task creator to handle the error in a timely manner.

Migration task management

A migration task is a basic unit of data migration. To migrate data, you must create a migration task in the DTS console. To create a migration task, you must configure the required information such as the source and destination instances, the migration types, and the objects to be migrated. You can create, manage, stop, and delete data migration tasks in the DTS console.

The following table describes the statuses of a data migration task.

Task status	Description	Available operations
Not Started	The migration task is configured but the precheck is not performed.	Perform the precheck and delete the migration task
Prechecking	A precheck is being performed.	Delete the migration task
Precheck Passed	The migration task has passed the precheck but is not started.	Start and delete the migration task
Migrating	Data is being migrated.	Pause, stop, and delete the migration task
Migration Failed	The migration task has encountered an error. You can identify the point of failure based on the progress of the migration task.	Delete the migration task
Paused	The migration task is paused.	Start and delete the migration task
Completed	The migration task is completed, or you have clicked End to stop the migration task.	Delete the migration task

If a migration task is in the **Migration Failed** state, the task will be started and resumed within seven days. To stop data migration, you must manually release or stop the migration task.

References

- To start data migration, you can log on to the DTS console to configure a data migration task.
- To migrate applications, you can use the Alibaba Cloud Migration Tool.

5.3. Change tracking (new)

You can use Data Transmission Service (DTS) to track data changes from user-created MySQL databases and ApsaraDB RDS for MySQL instances in real time. This feature applies to the following scenarios: cache updates, business decoupling, synchronization of heterogeneous data, and synchronization of extract, transform, and load (ETL) operations.

The previous change tracking feature requires high costs. To improve user experience and reduce usage costs, Alibaba Cloud has upgraded the change tracking feature to achieve repeated consumption of data across multiple downstream instances.

Features

- You can consume data across multiple downstream instances and filter databases and data tables. For more information, see Consumer groups.
- You can track data changes from user-created MySQL databases.
- You can use the change tracking feature in the classic network and VPCs. You can track and consume data within a VPC for lower network latency and higher data security.

> **Note** You can select a network type when you create a change tracking task. For more information, see Track data changes from an ApsaraDB RDS for MySQL instance.

- You can consume data in a change tracking task by using a Kafka client.

  > **Note** Only Kafka versions 0.10.0.x to 1.1.x are supported. For more information, see Use a Kafka client to consume tracked data.

- You can add or remove objects for change tracking. For more information, see Modify the objects for change tracking.
- You can modify consumption checkpoints.

  > **Note** You must use your change tracking client to modify consumption checkpoints.

- You can monitor the status of change tracking tasks. If the latency threshold for downstream consumption is reached, you will receive an alert. You can set the alert threshold based on the sensitivity of your business to consumption latency.

Consumer groups

When you use the previous change tracking feature, one downstream SDK client consumes data and other SDK clients act as backup clients. If you need to repeatedly consume incremental data in the same source database, you must purchase multiple change tracking instances. This incurs high costs.

DTS provides consumer groups to reduce your costs. You can create consumer groups for downstream consumers who need to subscribe to the same database instance. Consumers in these consumer groups can subscribe to the same database instance and consume data at the same time.

Consumer group features

- You can create multiple consumer groups in a change tracking instance to repeatedly consume data.
- A consumer group consumes each message only once. Each consumer in the consumer group serves as a backup for other consumers.
- In a consumer group, only one consumer can consume data at a time. Other consumers act as disaster recovery nodes.

Supported database types and versions

A source MySQL database can be one of the following types: **User-Created Database with Public IP Address**, **User-Created Database in ECS Instance**, and **User-Created Database Connected Over Express Connect, VPN Gateway, or Smart Access Gateway**.

Source database	Data change type
User-created MySQL database 5.1, 5.5, 5.6, 5.7, or 8.0	• Data update • Schema update
ApsaraDB RDS for MySQL All versions	

Supported programming languages

DTS allows clients to consume data by using the following programming languages:

- Java
- C/C++
- Python
- Go (Golang)
- Erlang
- .NET
- Ruby
- Node.js
- Proxy (such as HTTP REST)

You can consume data by using a Kafka client for a specific language. For more information, see Kafka clients. DTS provides an example of using the Kafka client for Java to consume data. For more information, see Use a Kafka client to consume tracked data.

> **Warning** If you use the official Kafka client to consume data, you must specify the username in the following format: `<Consumer group account>-<Consumer group ID>`, for example, `dtstest-dtsae******bpv`. Otherwise, the connection fails.

Supported data change types and objects

- Data change types

  Data changes include schema updates that use the data definition language (DDL) and data updates that use the data manipulation language (DML).

  - Schema update

    You can track the operations that create, delete, or modify the schemas of objects in an instance. To do this, you must use the change tracking client to filter the required data.

  - Data update

    You can track data updates of a selected object, including the INSERT, DELETE, and UPDATE operations.

- Objects for change tracking

  The objects for change tracking include tables and databases.

Change tracking tasks

A change tracking task pulls data changes from the source database in real time and stores the data changes that were generated in the last 24 hours. You can use a Kafka client to track and consume data in the change tracking task. You can create, manage, and delete change tracking tasks in the DTS console.

The following table describes the statuses of a change tracking task when you create and run the task.

Task status	Description	Available operation
Prechecking	The configuration of the change tracking task is complete and a precheck is being performed.	Delete the change tracking task
Not Started	The change tracking task has passed the precheck but has not been started.	• Start the change tracking task • Delete the change tracking task
Performing Initial Change Tracking	The initial change tracking is in progress. This process takes about 1 minute.	Delete the change tracking task
Normal	Data changes are being pulled from the data source.	• View the demo code • Configure monitoring and alerts • Delete the change tracking task • Reconfigure the change tracking task
Error	An error occurs when the change tracking task pulls data changes from the data source.	• View the demo code • Configure monitoring and alerts • Delete the change tracking task • Reconfigure the change tracking task

Note

- A change tracking task in the **Error** state will be resumed within seven days. To stop change tracking, you can delete the change tracking task.
- After a change tracking task is deleted, the data tracked by the task is cleared and cannot be recovered. We recommend that you use caution when you perform this operation.

5.4. Change tracking (previous)

You can use Data Transmission Service (DTS) to track data changes from ApsaraDB RDS for MySQL instances in real time. This feature applies to the following scenarios: cache updates, business decoupling, synchronization of heterogeneous data, and synchronization of extract, transform, and load (ETL) operations.

Supported instance types

Source database	Data change type
ApsaraDB RDS for MySQL Versions 5.5, 5.6, and 5.7 **Note** The previous change tracking feature is no longer available. We recommend that you use the new change tracking feature. For more information, see Change tracking (new).	• Data update • Schema update

Objects for change tracking

The objects for change tracking include databases and tables.

Data changes include the following types:

- Data update that uses data manipulation language (DML)
- Schema update that uses data definition language (DDL)

When you configure a change tracking task, you can select a data change type based on your business requirements.

Change tracking tasks

A change tracking task is the basic unit of change tracking and data consumption. To track data changes from a database instance, you must create a change tracking task in the DTS console for the instance.

The change tracking task pulls data changes from the database instance in real time and stores the data changes that were generated in the last 24 hours. You can use the DTS SDK to consume the tracked data. You can also create, manage, or delete change tracking tasks in the DTS console.

A change tracking task can be consumed by only one downstream SDK client. To track data changes from a database instance by using multiple downstream SDK clients, you must create an equivalent number of change tracking tasks. Database instances of these change tracking tasks share the same instance ID.

The following table describes the statuses of a change tracking task when you create and run the task.

Task status	Description	Available operation
Prechecking	The configuration of the change tracking task is complete and a precheck is being performed.	Delete the change tracking task

Task status	Description	Available operation
Not Started	The change tracking task has passed the precheck but has not been started.	• Start the change tracking task • Delete the change tracking task
Performing Initial Change Tracking	The initial change tracking is in progress. This process takes about 1 minute.	Delete the change tracking task
Normal	Data changes are being pulled from the data source.	• View the demo code • View the tracked data • Delete the change tracking task
Error	An error occurs when the change tracking task pulls data changes from the data source.	• View the demo code • Delete the change tracking task

A change tracking task in the **Error** state will be resumed within seven days. To stop change tracking, you must pause or stop the change tracking task.

Advanced features

You can use the following advanced features that are provided for change tracking:

- Add or remove the objects for change tracking

  You can add or remove the required objects when a change tracking task is running.

- View the tracked data

  In the DTS console, you can view the data that is tracked by a change tracking task.

- Modify consumption checkpoints

  You can modify the consumption checkpoints.

- Monitor change tracking tasks

  You can monitor the status of change tracking tasks. If the threshold for consumption delay is reached, you will receive an alert. You can set the alert threshold based on the sensitivity of your business to consumption delay.

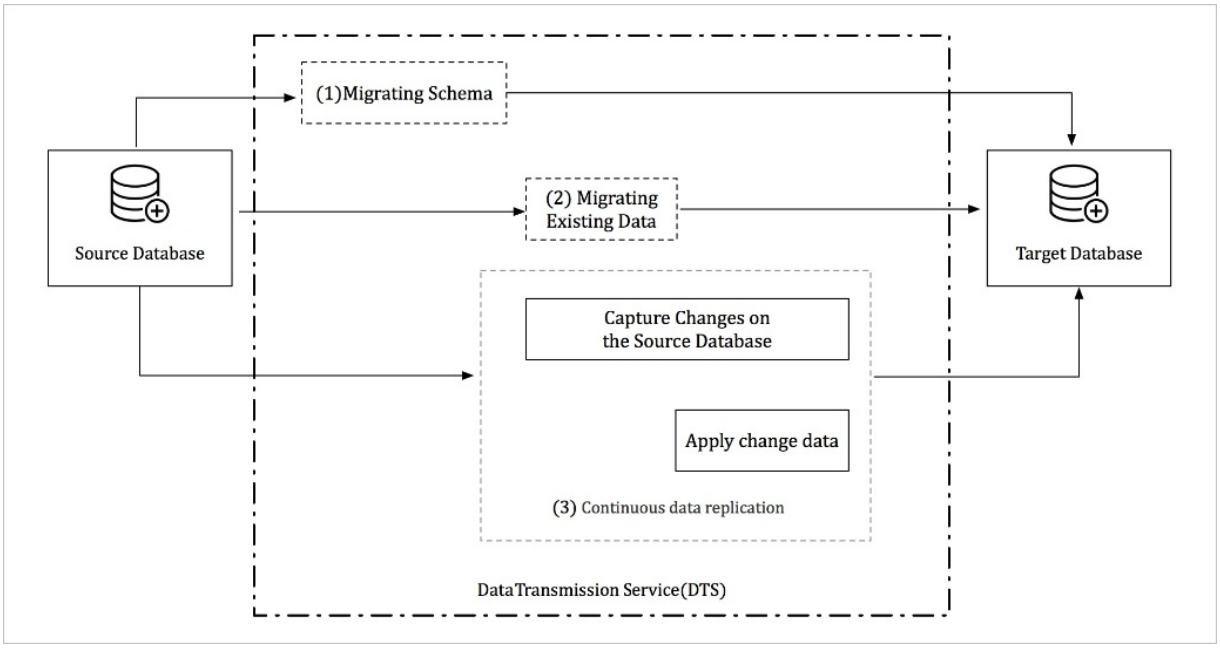
6.Scenarios

Data Transmission Service (DTS) supports data migration, change tracking, and real-time data synchronization in various scenarios.

Database migration with minimized downtime

To ensure data consistency, traditional migration requires that you stop writing data to the source database during data migration. Depending on the data volume and network conditions, the migration may take several hours or even days, which has a great impact on your businesses.

DTS provides migration with minimized downtime. Services are always available except when they are switched from the source instance to the destination instance. The service downtime is minimized to minutes. The following figure shows the architecture of data migration.

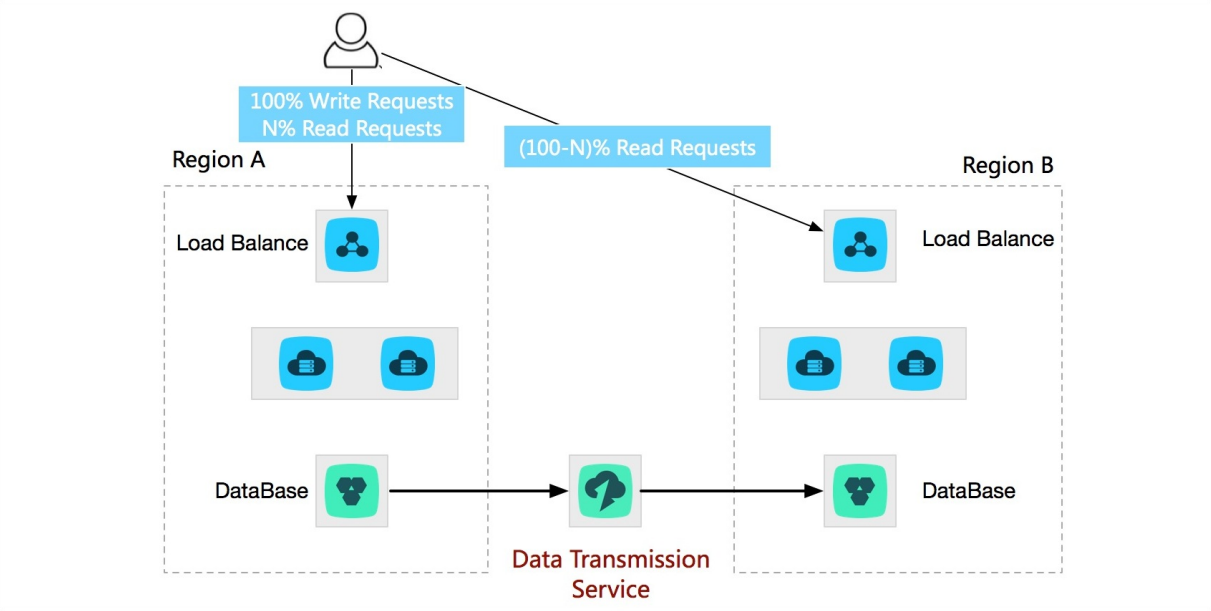


The data migration process includes schema migration, full data migration, and incremental data migration. During incremental data migration, the data in the source instance is synchronized to the destination instance in real time. You can verify businesses in the destination database. After the verification succeeds, you can migrate businesses to the destination database.

Geo-disaster recovery

If your businesses are deployed in a single zone, service interruption may occur because of force majeure factors such as power failure and network disconnection.

In this case, you can build a geo-disaster recovery center in another zone to ensure service availability. DTS synchronizes data between the geo-disaster recovery center and the business center in real time to ensure data consistency. If a failure occurs in the business center, you can switch the traffic to the geo-disaster recovery center.



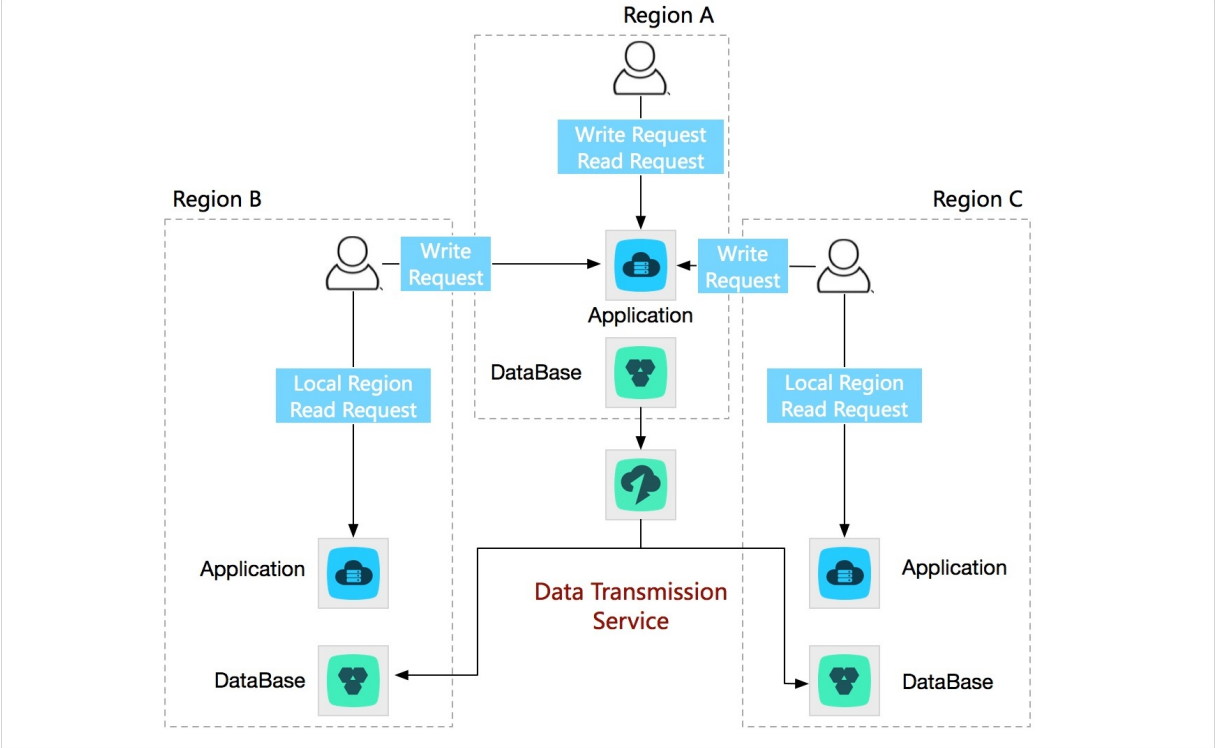
Active geo-redundancy

With the rapid development of businesses and the increase in the number of users, you may encounter the following issues if you deploy businesses in a single region:

- Users are distributed across a wide range of geographical locations, and distant users have high access latency, which affects user experience.
- The scalability is limited by the capacity of infrastructure in a single region, such as power supply and network bandwidth.

To solve the preceding issues, you can build multiple business units in the same city or different cities. DTS enables two-way real-time data synchronization between business units to ensure global data consistency. If a failure occurs in a business unit, you only need to switch the traffic of this business unit to another business unit. The businesses can be recovered within seconds. The redundancy of multiple business units ensures high availability.

You can also distribute traffic across business units based on a specific dimension. For example, you can reschedule the traffic of each business unit based on the region to allow users to access the nearest node. This reduces network latency and improves user experience. The scalability is no longer limited by the capacity of infrastructure because business units are distributed across different regions.



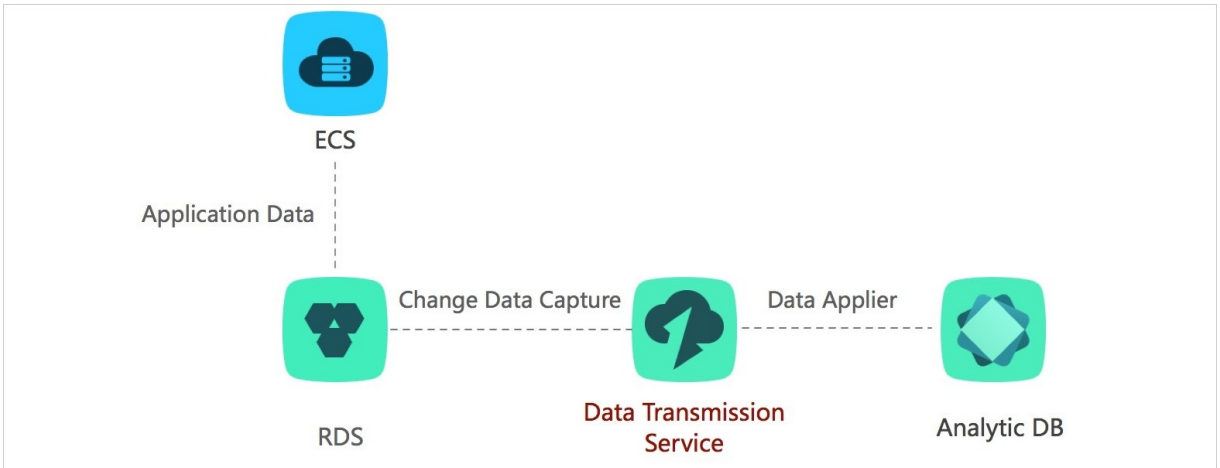
Custom BI system built with more efficiency

User-created BI systems cannot meet increasing requirements for real-time capability. Alibaba Cloud provides complete BI systems. DTS allows you to synchronize data in real time from user-created databases to Alibaba Cloud BI storage systems, such as MaxCompute. DTS helps you build a custom BI system that meets your business requirements on Alibaba Cloud.

Real-time data analysis

Data analysis is essential in improving enterprise insights and user experience. With real-time data analysis, enterprises can adjust marketing strategies to adapt to changing markets and higher demands for better user experience.

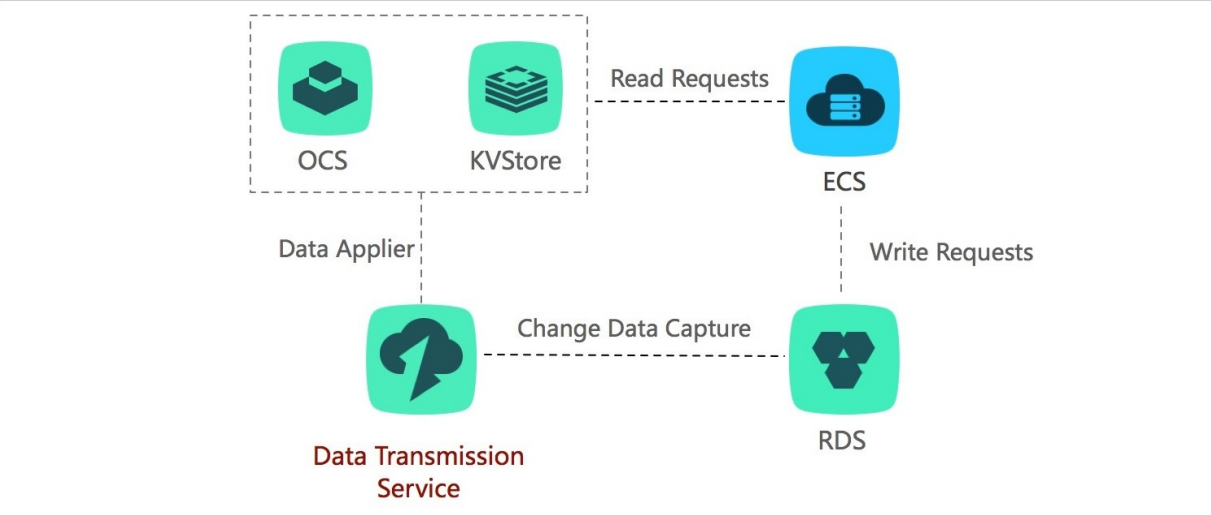
With the change tracking feature provided by DTS, you can acquire real-time incremental data without affecting online businesses. You can use the DTS SDK to synchronize the subscribed incremental data to the analysis system for real-time analysis.



Lightweight cache update policies

To accelerate access speed and improve concurrent read performance, a cache layer is used in the business architecture to receive all read requests. The memory read mechanism of the cache layer can help to improve read performance. The data in the cache memory is not persistent. If the cache memory fails, the data in the cache memory will be lost.

With the change tracking feature provided by DTS, you can subscribe to the incremental data in databases and update the cached data to implement lightweight cache update policies.



Benefits

- Quick update with low latency

  The business returns data after the database update is complete. For this reason, you do not need to consider the cache invalidation process, and the entire update path is short with low latency.

- Simple and reliable applications

  The complex doublewrite logic is not required for the applications. You only need to start the asynchronous thread to monitor the incremental data and update the cached data.

- Application updates without extra performance consumption

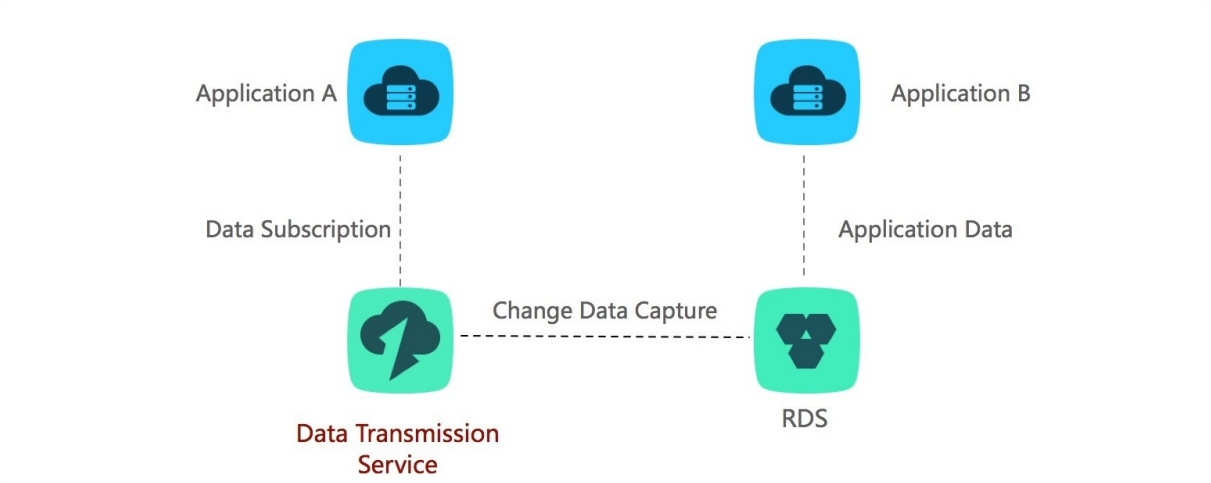
  DTS retrieves incremental data by parsing incremental logs in the database, which does not affect the performance of businesses and databases.

Business decoupling

The e-commerce industry involves many different types of business logic such as ordering, inventory, and logistics. If all of these types of business logic are included in the ordering process, the order result can be returned only after all the changes are complete. However, this may cause the following issues:

- The ordering process consumes a long period of time and results in poor user experience.
- The business system is unstable and downstream faults will affect service availability.

With the change tracking feature provided by DTS, you can optimize your business system and receive notifications in real time. You can decouple different types of business logic and asynchronously process data. This makes the core business logic simpler and more reliable. The following figure shows the architecture of business decoupling.



In this scenario, the ordering system returns the result after the buyer places an order. The underlying layer obtains the data changes that are generated in the ordering system in real time by using the change tracking feature. You can subscribe to these data changes by using the DTS SDK, which triggers different types of downstream business logic such as inventory and logistics. This ensures that the entire business system is simple and reliable.

This scenario has been applied to a wide range of businesses in Alibaba Group. Tens of thousands of downstream businesses in the Taobao ordering system are using the change tracking feature to retrieve real-time data updates and trigger business logic every day.

Scalable read capability

A single database instance may not have sufficient resources to deal with a large number of read requests. You can use the real-time synchronization feature of DTS to build read-only instances and distribute read requests across these read-only instances. This allows you to scale out the read capability and relieve the pressure on the primary database instance.

7.Specifications

7.1. Specifications of data synchronization instances

This topic describes the specifications of data synchronization instances and provides the results of performance testing.

Precautions

The performance metrics provided in this topic are used for reference only and are not used as criteria for SLA evaluation.

Terms

- Specification: DTS provides data synchronization instances that have different specifications. The performance of these instances depends on the number of synchronized records.
- RPS: the number of SQL statements that are synchronized per second, including BEGIN operation, COMMIT operation, DML statements (INSERT, UPDATE, and DELETE operations), and DDL statements.

Specifications of data synchronization instances

DTS provide the following four specifications based on the maximum performance of data synchronization instances: micro, small, medium, and large. The data synchronization instance of each specification can reach the maximum performance if the following conditions are met:

- The pressure on the source instance must be greater than or equal to the maximum performance that corresponds to each specification.
- The destination instance does not have bottlenecks in write performance and supports the performance pressure that corresponds to each specification.
- The network latency between the DTS server and the source or destination instance does not exceed 2 milliseconds.

Specification	Maximum performance (RPS)
micro	Less than 200
small	200 to 2,000
medium	2,000 to 5,000
large	Unlimited **Note** The actual performance of the large specification depends on the network environment and the performance of the source and destination instances.

7.2. Specifications of data migration instances

This topic describes the specifications of data migration instances and provides the results of performance testing.

Precautions

The performance metrics provided in this topic are used only for reference and are not used as criteria for product SLA evaluation.

Terms

Term	Description
specification	DTS provides data migration instances that have different specifications. The performance of these instances depends on the performance of incremental data migration.
table quantity	The total number of tables in the test model.
record size	The size of each record that is migrated during incremental data migration.
RPS	The number of records per second (RPS) that are changed by INSERT, UPDATE, and DELETE operations in the source database.

> **Note**
>
> - If an SQL statement contains operations on multiple rows of data, DTS identifies the operations as multiple data changes. If you perform INSERT, UPDATE, and DELETE operations on a data record multiple times, DTS also identifies the operations as multiple data changes.
> - DTS identifies each COMMIT operation as a data change.

Specifications

DTS provide the following four specifications based on the maximum performance of data migration instances: small, medium, large, and 2xlarge. The data migration instance of each specification can reach the maximum performance if the following conditions are met:

- The pressure on the source instance must be greater than or equal to the maximum performance that corresponds to each specification.
- The destination instance does not have bottlenecks in write performance and supports the performance pressure that corresponds to each specification.
- The network latency between the DTS server and the source or destination instance does not exceed 2 milliseconds.

Specification	Maximum performance (RPS)
small	200 to 2,000

Specification	Maximum performance (RPS)
medium	2,000 to 5,000
large	Unlimited
2xlarge	Note The online running performance of the large specification depends on the network environment and the performance of the source and destination instances.

Test model

Test procedure: Create an incremental migration task between two ApsaraDB RDS for MySQL instances. Then, perform a stress test on the source ApsaraDB RDS for MySQL instance to view the performance of incremental data migration.

Test environment

Instance	RDS instance configuration	Maximum performance
Source instance	• Instance specification: rds.mys2.8xlarge • Memory: 48,000 MB • Maximum connections: 2,000	• Maximum QPS: 18,000 • Maximum IOPS: 14,000
Destination instance	• Instance specification: rds.mys2.8xlarge • Memory: 48,000 MB • Maximum connections: 2,000	• Maximum QPS: 18,000 • Maximum IOPS: 14,000

Test model:

- The number of test tables is 20.
- Each test table has a primary key.
- The record size is 1 KB.
- Each transaction has an average of two DML operations and one COMMIT operation. The ratio of INSERT, UPDATE, and DELETE operations is 3:1:2.

Test results

Source instance region	Destination instance region	Network latency between instances (milliseconds)	Specification	RPS
China (Hangzhou)	China (Hangzhou)	0.26	small	2,566
China (Hangzhou)	China (Hangzhou)	0.26	medium	4,726
China (Hangzhou)	China (Hangzhou)	0.26	large	6,378

Source instance region	Destination instance region	Network latency between instances (milliseconds)	Specification	RPS
China (Hangzhou)	China (Qingdao)	26	small	2,469
China (Hangzhou)	China (Qingdao)	26	medium	4,856
China (Hangzhou)	China (Qingdao)	26	large	5,439
China (Hangzhou)	China (Beijing)	26	small	2,533
China (Hangzhou)	China (Beijing)	26	medium	5,038
China (Hangzhou)	China (Beijing)	26	large	6,829
China (Hangzhou)	US (Silicon Valley)	175	small	1,753
China (Hangzhou)	US (Silicon Valley)	175	medium	2,837
China (Hangzhou)	US (Silicon Valley)	175	large	3,884
Singapore (Singapore)	US (Silicon Valley)	198	small	1,104
Singapore (Singapore)	US (Silicon Valley)	198	medium	1,724
Singapore (Singapore)	US (Silicon Valley)	198	large	2,256

Note The preceding test results show the maximum performance of data migration instances that are configured with different specifications. The performance of incremental data migration cannot be guaranteed in the following cases: The table to be migrated does not have a primary key, the network latency is high, an update hotspot exists, or the source and destination instances have performance bottlenecks.

8.Terms

This topic describes the terms that are used in the DTS documentation.

Term	Description
precheck	The system performs a precheck before starting a data migration task, data synchronization task, or change tracking task. The following items are checked: the connectivity between the DTS server and the source and destination databases, database account permissions, whether binary logging is enabled, and database version numbers. **Note** If the precheck fails, click the icon next to each failed item to view the related details. Fix the issues as instructed and run the precheck again.
schema migration	DTS migrates the schemas of the objects that you have selected from the source instance to the destination instance. Tables, views, triggers, and stored procedures can be migrated. For schema migration between heterogeneous databases, DTS converts the schema syntax based on the syntax of the source and destination databases. For example, it converts the NUMBER data type of Oracle databases to the DECIMAL data type of MySQL databases.
full data migration	DTS migrates historical inventory data of the objects that you have selected from the source database to the destination database. If you select only schema migration and full data migration, new data generated in the source database will not be migrated to the destination database. To ensure data consistency, we recommend that you do not write new data into the source database during full data migration. **Note** To migrate data with minimized downtime, you must select schema migration, full data migration, and incremental data migration.
incremental data migration	DTS synchronizes the incremental data generated in the source database to the destination database in real time. **Note** During incremental data migration, data between the source and destination databases is synchronized in real time. The migration task does not automatically end. You need to manually end the migration task.

Term	Description
initial synchronization	DTS synchronizes the schemas and historical inventory data of the objects that you have selected to the destination database before synchronizing incremental data. Initial synchronization includes initial schema synchronization and initial full data synchronization. • Initial schema synchronization: synchronizes the schemas of the required objects from the source database to the destination database. • Initial full data synchronization: synchronizes the historical inventory data of the required objects from the source database to the destination database.
synchronization performance	The synchronization performance is the number of records that are synchronized to the destination database per second. The unit is records per second (RPS). For more information, see Specifications of data synchronization instances.
synchronization latency	The synchronization latency is the difference between the timestamp of the latest data synchronized to the destination database and the current timestamp of the source database. Synchronization latency reflects the time difference between the source and destination databases for the latest data. If the synchronization latency is zero, data in the source database is consistent with that in the destination database.
data update	Data updates are operations that only modify data without modifying the schema, such as INSERT, DELETE, and UPDATE operations.
schema update	Schema updates are operations that modify the schema syntax, such as CREATE TABLE, ALTER TABLE, and DROP VIEW operations.
timestamp range	The timestamp range is the range of timestamps for the incremental data that is stored in the change tracking channel. By default, the change tracking channel retains the data that is generated in the most recent 24 hours. DTS regularly cleans expired incremental data and updates the timestamp range of the change tracking channel. **Note** The timestamp of the incremental data is the timestamp when the incremental data is updated in the source database and written into the transaction log.
consumption checkpoint	The consumption checkpoint is the timestamp of the latest incremental data that is consumed by using the downstream SDK client. Each time the SDK client consumes a data record, it returns a confirmation message to DTS. DTS updates and saves the consumption checkpoint. If the SDK client restarts due to exceptions, DTS automatically pushes subscribed data from the last consumption checkpoint.

9.FAQ

This topic answers frequently asked questions (FAQ) for Data Transmission Service (DTS).

Which databases does DTS support in data migration, data synchronization, and change tracking?

DTS supports data transmission between various data sources, such as relational database management systems (RDBMS), NoSQL databases, and online analytical processing (OLAP) databases. For more information, see Supported databases.

> **Note** DTS also supports data migration and synchronization from databases that are provided by third-party cloud vendors to Alibaba Cloud. For more information, see Overview of data migration scenarios.

How does data migration, data synchronization, and change tracking work?

For more information, see System architecture and design concepts.

What are the differences between data migration and data synchronization?

Item	Data migration	Data synchronization
Scenarios	The data migration feature is used to migrate data from on-premises databases, user-created databases hosted on ECS, and databases on third-party cloud platforms to Alibaba Cloud. You can release a data migration task immediately after it is complete.	The data synchronization feature is used to synchronize data between two data sources in real time. It is suitable for scenarios such as active geo-redundancy, disaster recovery, cross-border data synchronization, query load balancing, cloud BI systems, and real-time data warehousing. After you create a data synchronization task, it continuously synchronizes data between the specified data sources to ensure data consistency.
Supported databases	For more information, see Supported databases.	For more information, see Supported databases.
Supported deployment locations of databases	• User-created database with a public IP address • User-created database hosted on ECS • User-created database connected over Express Connect, VPN Gateway, or Smart Access Gateway	• User-created database hosted on ECS • User-created database connected over Express Connect, VPN Gateway, or Smart Access Gateway **Note** The data synchronization feature ensures low network latency through data transmission over the internal network.
Features	You can read and migrate data across user-created databases that are deployed on VPCs owned by different Alibaba cloud accounts.	• You can change the names of objects such as columns, tables, and databases. • You can filter the data to be synchronized. • You can modify the objects to be synchronized. • You can configure two-way data synchronization between MySQL databases. • You can select the types of operations to be synchronized. For example, you can synchronize only INSERT operations.
Billing methods	Only the pay-as-you-go billing method is supported.	The pay-as-you-go and subscription billing methods are supported.

Item	Data migration	Data synchronization
Billing rules	You are billed when incremental data migration is in progress, including the period when incremental data migration is paused. You are not billed for schema migration and full data migration.	• If you use the pay-as-you-go billing method, billing starts after the task is configured and started. You are also billed when the task is paused. • If you use the subscription billing method, the fee is deducted at the time of purchase based on the quantity and configurations you select.

Note For databases that are not supported by the data synchronization feature (such as databases on ApsaraDB for MongoDB instances), you can perform incremental data migration. Incremental data migration can ensure that data is synchronized between the source and destination database.

How is the synchronization delay measured?

The synchronization delay is the difference between the timestamp of the latest synchronized data in the destination database and the current timestamp in the source database. The unit is milliseconds.

What can I do if DTS cannot connect to the database?

For more information, see Source database connectivity and Destination database connectivity.

What is the billing method of DTS?

For more information, see Pricing.

What are the differences between different channel specifications?

For more information, see Specifications of data migration instances and Specifications of data synchronization instances.

Can channel specifications be downgraded?

No, channel specifications cannot be downgraded.

Why is the price of data synchronization higher than that of data migration?

Data synchronization comes with more advanced features. For example, you can modify the objects to be synchronized. You can configure two-way data synchronization between MySQL databases. In addition, the data synchronization feature ensures low network latency through data transmission over the internal network.

Can I migrate or synchronize data across Alibaba Cloud accounts?

- Data migration: You can migrate data across Alibaba Cloud accounts between ApsaraDB RDS for MySQL instances. For more information, see Migrate data between RDS instances under different Alibaba Cloud accounts. You can also migrate data across Alibaba Cloud accounts between other database instances, such as Apsara PolarDB for MySQL, DRDS, Redis, and MongoDB. To do this, you can specify the database as a **user-created database with a public IP address** when you

configure the data migration task.

- Data synchronization: You can synchronize data across Alibaba Cloud accounts only between ApsaraDB RDS for MySQL instances. For more information, see Synchronize data between ApsaraDB RDS for MySQL instances that belong to different Alibaba Cloud accounts.

Can I migrate data within a single instance?

Yes, you can migrate data within a single instance. For more information, see Migrate data between databases that have different names.

Can I migrate or synchronize DML and DDL operations?

Yes, you can migrate or synchronize DML and DDL operations between relational databases. The supported DML operations are INSERT, UPDATE, and DELETE. The supported DDL operations are CREATE, DROP, ALTER, RENAME, and TRUNCATE.

> **Note** The supported DML and DDL operations are different in different scenarios. For example, if you synchronize data from a MySQL database to AnalyticDB for MySQL 2.0, only the following DDL operations are supported: CREATE TABLE, ALTER TABLE, and DROP TABLE. Only the following DML operations are supported: INSERT, UPDATE, and DELETE. For more information, see the topics about data migration and synchronization scenarios.

Can I migrate or synchronize database shards and table shards?

Yes, you can migrate or synchronize database shards and table shards. For example, you can migrate or synchronize database shards and table shards from a MySQL database and Apsara PolarDB for MySQL cluster to AnalyticDB for MySQL. This allows you to merge multiple tables.

Can I migrate or synchronize data across time zones and character sets?

Yes, you can migrate or synchronize data across time zones and character sets.

Can I change the name of objects that are migrated or synchronized to the destination database?

Yes, you can change the name of columns, tables, and databases by using the object name mapping feature. For more information, see Object name mapping and Rename an object to be synchronized.

Can I filter fields or data?

Yes, you can filter fields or data in a table. For more information, see Filter the data to be migrated and Use SQL conditions to filter data.

Can I add or remove objects to be synchronized?

Yes, you can add or remove objects to be synchronized. For more information, see Add an object to a data synchronization task and Remove an object from a data synchronization task.

How do I view the performance of data migration and synchronization?

For more information, see View the connection status and performance of incremental data migration and View the connection status and performance of data synchronization.

How do I consume tracked data?

- Change tracking (previous): You can use the SDK to consume tracked data. For more information, see Use the SDK to consume tracked data.

  > **Note** Only the SDK for Java is supported. For more information about SDK versions, see Download the SDK.

- Change tracking (new): You can use a Kafka client to consume tracked data. For more information, see Use a Kafka client to consume tracked data.