

DDT Application Scenarios

Version 1.0.0

Sep.13,2024

Table of Contents

DDT Application Scenarios	3
Startup Conditions	3
Startup Conditions:	3
Functionality:	3
Starting DDT + Monitoring	3
Startup Conditions:	4
Functionality:	4
Data Verification Module Launched Separately	5
Startup Conditions:	5
Functionality:	5
Full Data Synchronization Benchmark Results	6
Cache Settings:	6
Benchmark Results:	6
Real-Time Synchronization Benchmark Results	7
Cache Settings:	7
Benchmark Results:	7
Advantages:	8
Summary:	8

DDT Application Scenarios

Startup Conditions

Startup Conditions:

Starting DDT with only the main module is suitable for general document data transfer tasks where real-time monitoring or post-transfer verification is not necessary. This configuration is ideal for environments with limited system resources or when the primary goal is to complete data transfer tasks efficiently without the additional overhead of monitoring or verification processes.

Functionality:

- **Efficient Data Transfer:** The core functionality of the main DDT module is to facilitate the seamless transfer of document data between source and target databases. It supports various data architectures and scales, using optimized algorithms to ensure swift transfer while maintaining data integrity and consistency. The focus on core data transfer tasks ensures that operations are streamlined for optimal performance.
- **Simplified Operation:** This mode is designed for ease of use, making it straightforward to configure and initiate. It is particularly beneficial for tasks that do not demand real-time oversight or detailed verification. The simplified approach minimizes deployment and maintenance complexity, allowing users to focus on the core data transfer process without additional setup requirements.
- **Performance Optimization:** By dedicating system resources exclusively to the data transfer task, this mode enhances performance. The absence of additional monitoring or verification functions results in more efficient use of system resources, leading to faster processing speeds and improved overall efficiency during data transfer operations.

Starting DDT + Monitoring



Startup Conditions:

This mode involves launching DDT along with an additional monitoring module. It is designed for scenarios where real-time monitoring of the data transfer process is crucial. This configuration is particularly valuable for high-stakes data transfers or when transparency and immediate feedback are essential.

Functionality:

- Real-Time Monitoring and Visualization:** The monitoring module works in conjunction with the main DDT module to provide real-time updates on key performance indicators. Users can view metrics such as transfer speed, error rates, and latency through graphical representations, which offer a clear view of the data transfer process. This visual feedback is critical for managing and optimizing the transfer in real-time.
- Real-Time Alerts and Log Analysis:** The monitoring module is equipped with real-time alerting capabilities, notifying users of anomalies such as elevated error rates or significant delays. These alerts enable prompt issue resolution. Additionally, post-transfer log analysis helps in identifying and addressing any issues that arose during the transfer, ensuring that all problems are resolved for future operations.
- Enhanced Traceability:** For scenarios that require detailed tracking of the data transfer process, the monitoring module provides comprehensive logging and reporting features. This allows users to trace every stage of the transfer, enhancing transparency and

control over the synchronization process, and ensuring that all steps are documented and reviewable.

Data Verification Module Launched Separately



Startup Conditions:

In this mode, the data verification module is launched independently, typically after the completion of the data transfer. This configuration is ideal for scenarios where rigorous verification of data integrity and consistency is necessary. It is crucial for ensuring that the transferred data is accurate and complete, especially in sensitive or critical applications.

Functionality:

- **Precise Data Verification:** The data verification module conducts thorough checks to ensure that the data in the target database matches the source database precisely. It employs various verification techniques, including hash comparisons and row count checks, to validate the accuracy and completeness of the data. This detailed approach ensures that no discrepancies exist between the source and target datasets.
- **Protection Against Data Loss and Tampering:** By performing meticulous verification, this module can detect and address any issues related to data loss, duplication, or tampering that may have occurred during the transfer. This added layer of scrutiny enhances data security, ensuring that the data remains intact and unaltered throughout the migration process.

- **Suitable for High-Precision Tasks:** The data verification module is especially beneficial for tasks requiring high accuracy post-transfer, such as financial data migrations or medical record synchronization. It guarantees that all data is transferred correctly, meeting the stringent reliability and security requirements typical of these critical applications.

Full Data Synchronization Benchmark Results

Cache Settings:

- cacheBucketSize: 64
- cacheBucketNum: 64
- dataBatchSize: 128
- cacheSize: 30GB

Benchmark Results:

Source Read Threads	Target Write Threads	QPS	Memory Usage	CPU Utilization
5	15	370,042	25,159 MB	812%
6	20	390,000	26,522 MB	1080%
6	24	400,138	26,655 MB	1160%
8	24	360,209	25,252 MB	1120%

Advantages:

- **High Data Transfer Performance:** The system demonstrated a peak performance of 400,138 QPS (Queries Per Second) under different configurations. This reflects the system's robust processing power and efficiency in handling full data synchronization tasks.
- **Optimized Performance Configuration:** By adjusting thread counts, the system's QPS improved significantly. The configuration with 6 source read threads and 24 target write threads achieved the highest QPS, showcasing the benefits of optimized resource allocation.
- **Efficient Resource Utilization:** Despite increased CPU and memory usage at higher loads, the system effectively managed resources to ensure smooth data transfer. This high utilization indicates the system's capability to handle large-scale synchronization while maintaining performance stability.

Real-Time Synchronization Benchmark Results

Cache Settings:

- cacheBucketSize: 16
- cacheBucketNum: 16
- dataBatchSize: 128

Benchmark Results:

Oplog NS	Oplog Write	Oplog Read	Oplog NS Bucket	QPS	CPU Utilization	Memory Usage
1	6	1	2	72,398	280%	8,258 MB

1	9	1	3	80,385	240%	14,418 MB
1	12	1	4	79,365	280%	15,728 MB
1	15	1	5	75,388	280%	14,025 MB

Advantages:

- **Outstanding Real-Time Processing Capability:** The system reached a peak of 80,385 QPS under various oplog write configurations, demonstrating strong performance and efficiency in handling real-time data synchronization tasks.
- **Efficient Resource Management:** Even with high CPU and memory demands, the system managed resources effectively to ensure smooth and accurate real-time synchronization. The stable CPU utilization up to 280% reflects the system's reliability under continuous data operations.
- **Optimized Cache Configuration:** The use of smaller cache settings improved response times and data processing efficiency for real-time tasks. This optimized cache configuration contributed to faster and more efficient data synchronization.

Summary:

The detailed performance benchmarks highlight DDT's exceptional capabilities in both full data and real-time synchronization scenarios. In full data synchronization, the system demonstrated high query performance and efficient resource use, making it well-suited for extensive data transfer tasks. For real-time synchronization, DDT exhibited impressive processing power a

nd effective resource management, handling high-frequency data operations with stability. These results underscore the system's robust performance and versatility in various data synchronization applications, ensuring reliable and efficient operations across different use cases.