

# IBM Integrated Synchronization Incremental Updates Unleashed

Christian Michel

Cüneyt Göksu

Günter Schöllmann



**z Systems**





## IBM Integrated Synchronization: Incremental Updates Unleashed

The IBM® Db2® Analytics Accelerator (Accelerator) is a logical extension of Db2 for IBM z/OS® that provides a high-speed query engine that efficiently and cost-effectively runs analytics workloads. The Accelerator is an integrated back-end component of Db2 for z/OS. Together, they provide a hybrid workload-optimized database management system that seamlessly manages queries that are found in transactional workloads to Db2 for z/OS and queries that are found in analytics applications to Accelerator. Each query runs in its optimal environment for maximum speed and cost efficiency.

The incremental update function of Db2 Analytics Accelerator for z/OS updates Accelerator-shadow tables continually. Changes to the data in original Db2 for z/OS tables are propagated to the corresponding target tables with a high frequency and a brief delay. Query results from Accelerator are always extracted from recent, close-to-real-time data.

An incremental update capability that is called IBM InfoSphere® Change Data Capture (InfoSphere CDC) is provided by IBM InfoSphere Data Replication for z/OS up to Db2 Analytics Accelerator V7.5. Since then, an extra new replication protocol between Db2 for z/OS and Accelerator that is called IBM Integrated Synchronization was introduced. With Db2 Analytics Accelerator V7.5, customers can choose which one to use.

IBM Integrated Synchronization is a built-in product feature that you use to set up incremental updates. It does not require InfoSphere CDC, which is bundled with IBM Db2 Analytics Accelerator.

In addition, IBM Integrated Synchronization has more advantages:

- ▶ Simplified administration, packaging, upgrades, and support. These items are managed as part of the Db2 for z/OS maintenance stream.
- ▶ Updates are processed quickly.
- ▶ Reduced CPU consumption on the mainframe due to a streamlined, optimized design where most of the processing is done on the Accelerator. This situation provides reduced latency.
- ▶ Uses IBM Z® Integrated Information Processor (zIIP) on Db2 for z/OS, which leads to reduced CPU costs on IBM Z and better overall performance data, such as throughput and synchronized rows per second.

- ▶ On z/OS, the workload to capture the table changes was reduced, and the remainder can be handled by zIIPs.
- ▶ With the introduction of an enterprise-grade Hybrid Transactional Analytics Processing (HTAP) enabler that is also known as the *Wait for Data* protocol, the integrated low latency protocol is now enabled to support more analytical queries running against the latest committed data.

IBM Db2 for z/OS Data Gate simplifies delivering data from IBM Db2 for z/OS to IBM Cloud® Pak® for Data for direct access by new applications. It uses the special-purpose integrated synchronization protocol to maintain data currency with low latency between Db2 for z/OS and dedicated target databases on IBM Cloud Pak for Data.

The following topics are described in this paper:

- ▶ “Use cases that bring business value to IBM Integrated Synchronization”
- ▶ “Introduction to IBM Integrated Synchronization” on page 4
- ▶ “Benchmarks, CPU and elapsed time metrics, and comparisons with InfoSphere CDC” on page 6
- ▶ “Comparing InfoSphere Change Data Capture and IBM Integrated Synchronization” on page 8
- ▶ “Different scenarios for data sharing” on page 10
- ▶ “Setup details, prerequisites and recommended maintenance, and installation and configuration” on page 16
- ▶ “Prerequisites and recommended maintenance” on page 16
- ▶ “Security setup in RACF” on page 18
- ▶ “Security setup for network” on page 24
- ▶ “Network setup for data sharing groups” on page 29
- ▶ “Db2 setup” on page 30
- ▶ “Accelerator setup for incremental updates” on page 32
- ▶ “Installation experiences from other customers” on page 34
- ▶ “Enabling tables for replication” on page 35
- ▶ “Replicated tables in error” on page 35
- ▶ “IBM Integrated Synchronization externals in Db2 for z/OS” on page 35

# Use cases that bring business value to IBM Integrated Synchronization

The following use cases bring new capabilities to the incremental update capability of IBM Db2 Analytics Accelerator or provide new ways to enhance business needs.

## Migrating from InfoSphere CDC to IBM Integrated Synchronization

Migration from InfoSphere CDC to IBM Integrated Synchronization provides advantages over InfoSphere CDC. Because of the IBM Integrated Synchronization lightweight capture design compared to InfoSphere CDC, it provides better replication throughput (inserts, updates, and deletes per second). The lower latency of IBM Integrated Synchronization (the time until the update occurs in Db2 for z/OS is visible on Accelerator) compared to InfoSphere CDC (30 seconds compared to 2 seconds) provides recent data access on Accelerator.

With IBM Integrated Synchronization, workloads use less CPU on z/OS. If a workload on IBM Integrated Synchronization is eligible to be offloaded to zIIP (InfoSphere CDC does not have zIIP eligible workloads), then CPU is further reduced. These qualities make IBM Integrated Synchronization much more cost-effective compared to InfoSphere CDC. InfoSphere CDC has a capture agent on z/OS and stages captured logs for large uncommitted transactions both in memory and L2 cache in Virtual Storage Access Method (VSAM). IBM Integrated Synchronization transmits captured logs without any staging to Accelerator, and all staging processes happen on Accelerator so that IBM Integrated Synchronization uses less memory on z/OS compared to InfoSphere CDC.

Another area for this use case is simplified maintenance. The log reader component that is part of Db2 for z/OS receives maintenance updates as part of the Db2 for z/OS maintenance instead of separate InfoSphere CDC function modification identifiers (FMIDs). This situation simplifies the maintenance operation of the solution, which requires no extra InfoSphere CDC skills. This situation also reduces complexity by using a simplified installation.

## Migrating from a partial reload to an incremental update by using IBM Integrated Synchronization

Some customers cannot afford to use InfoSphere CDC based incremental update because of CPU costs, more required skills, latency, and other reasons, so they use partial reloads or microbatches to keep Accelerator updated. With IBM Integrated Synchronization, they receive the following benefits:

- ▶ Reduced latency provides changes on Accelerator within a few seconds so that customers who can afford minimum latency work with the most committed, near real-time data.
- ▶ Because there is minimum need to run partial reload jobs, the maintenance cost of those jobs or troubleshooting requirements are at a minimum. The CPU cost of unload jobs to z/OS is removed, but the log reading cost of integrated synchronization, which is zIIP eligible, is added to Db2 for z/OS.
- ▶ Although the incremental update per row is slower than partial reload, partial reload can transfer millions of unchanged rows many times. In this case, replication can be preferable to a partial reload.

## Workload that requires current and committed data from Db2 for z/OS

An important business requirement is to access and use the most current data from the source system. In large organizations, the IT department manages Db2 for z/OS and Accelerator, and lines of business (LOBs) manage the applications and workloads. The DBA or technical team are not fully aware of *data currency* requirements from the business, and sometimes this situation is an inhibitor to putting more workload on Accelerator.

With IBM Integrated Synchronization and the Wait for Data protocol, it is possible to access and use the most current data from the source system, and the application does not see a difference except in performance. Normally, the changes are not immediately visible to queries that are routed to Accelerator. However, some applications require that any change committed on Db2 is seen by a subsequent query. The Wait for Data protocol ensures that this situation occurs. Wait for Data means that analytics are performed against *committed* data that is known to be current relative to the SQL, and latency does not impact the consistency of the SQL results.

A good example of the Wait for Data protocol is Operational Data Store (ODS). While online transaction processing (OLTP) transactions update Db2 for z/OS, ODS queries that are enabled with the Wait for Data protocol run on Accelerator with IBM Integrated Synchronization. In this solution, ODS queries access the most committed current data on Accelerator.

## Introduction to IBM Integrated Synchronization

IBM Integrated Synchronization is an integrated, low-latency data coherence protocol between Db2 for z/OS and the Db2 Warehouse engine inside the Accelerator. It is zIIP enabled and has complete application transparency with an enterprise-grade HTAP enabler.

Figure 1 shows the IBM Integrated Synchronization architecture.

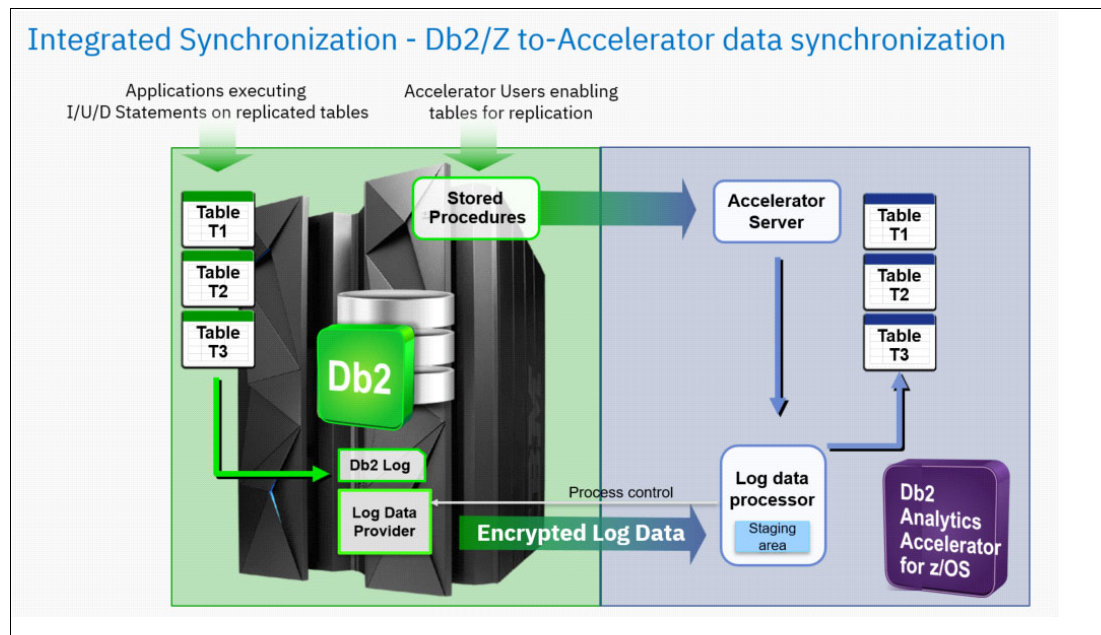


Figure 1 IBM Integrated Synchronization architecture

The tighter integration with Db2 for z/OS means that there is no extra product to configure and manage, much lower z/OS CPU and memory consumption, and log reading is zIIP eligible. IBM Integrated Synchronization supports a transactional consistency protocol that ensures that queries that are run by the Accelerator return the most recently committed data, which is the cornerstone of application transparency and HTAP.

IBM Integrated Synchronization works with the following products as follows:

- On Db2 for z/OS

The log data provider is a new internal Db2 for z/OS component that is provided with Db2 12 APAR PH06628 (Program Temporary Fix (PTF) UI63356). The log data provider is also known as the log reader task in Db2 for z/OS, and both terms can be used interchangeably. The log data provider is not a separate component to install and configure, but is part of the Db2 for z/OS engine. It adheres to the Db2 management lifecycle, which results in simplified management and support compared to InfoSphere CDC.

Log reading takes place through a service request block (SRB), which is an asynchronous process that is zIIP eligible. An SRB is a control block that represents a routine that performs a function or service in a specified address space. An SRB is like a task control block (TCB) in that it identifies a unit of work to the system. This SRB process runs within Db2 address space DBM1, and communication between Db2 for z/OS and Accelerator happens through the Db2 DIST address space's distributed data facility (DDF) port. Log data traffic is always encrypted through SSL, and it requires an AT-TLS setup on the DDF port in Db2 for z/OS.

When IBM Integrated Synchronization connects to Db2 for z/OS, it requires a user ID for authentication and to verify authorization to access the interface and the log records. Authentication with the z/OS user and password on a special IBM RACF® profile (ACCEL) is required. An RACF PassTicket is an alternative to using a password. Although any existing user can be used for IBM Integrated Synchronization, it is a best practice to use a dedicated user. After you create a user ID, it can be used either with a password or a PassTicket, that is, you can use either one when you authenticate.

There is no parsing, grouping, or any other processing of Db2 log records under Db2 for z/OS. It requests a subset of log records instead of reading all the logs by filtering the log records by tables that are synchronized to Accelerator.

- On Accelerator

Log data processor is a new internal Accelerator component that provides higher throughput and lower latency. There are unique performance optimizations compared to InfoSphere CDC, such as *bulk versus trickle* mode and optimistic apply. Bulk apply is ideal for a small set of tables with large changes, and it provides high throughput for a large batch of data. Trickle apply is ideal for many tables with small changes with consistent execution time. Log data processor dynamically switches between bulk and trickle apply modes. Bulk apply can process more than 1 million rows per second, and trickle apply can process more than 400 tables per second. Changes are parallelized in the back-end operations for better throughput. Microbatches reduce latency. A batch usually contains changes for multiple tables, and there is a sort table changes process that has priority.

The log data processor is an Accelerator specific *apply stack* with no general-purpose replication. The process is the key for the optimized design between Db2 for z/OS and Accelerator.

The changes that arrive on Accelerator are applied immediately, unlike in InfoSphere CDC or any other replication products, where only committed data is applied. The process that immediately applies changes is called *presumed commit*, which occurs in the early apply phase. When there is a rollback on the source, the changes are roll backed on the target too. The immediate apply is done for performance purposes to apply quickly a large amount of data within one transaction, but the user sees only committed data in the queries.

## Benchmarks, CPU and elapsed time metrics, and comparisons with InfoSphere CDC

This section provides benchmarks about IBM Integrated Synchronization resource usage. These benchmarks came from the lab environment that was used in the writing of this paper, so your results might be different.

### Test environment

Here are the components of the test environment that provided the benchmarks in this paper:

- ▶ Six z14 general-purpose central processors (GPCs) and two ZIIPs
- ▶ 96 GB of memory
- ▶ 10 Gb network to the Accelerator
- ▶ z/OS V02.02.00
- ▶ Db2 for z/OS V12
- ▶ M4002-010 (IIAS 1-rack)

### Test workload and test case

Here are the test workload and test case for the test environment:

- ▶ There 500 partitioned tables with concurrent insert/updates.
- ▶ Maximum Db2 transaction rates with a distributed Java test case:
  - 64.4 K rows per second inserted
  - 64.4 K rows per second updated
- ▶ Capture and apply processing happen with concurrent Db2® transactions.

Figure 2 on page 7 shows the concurrent Db2 transactions for this test workload and test case.



	CDC			Integrated Synchronization		
	CP	IIP	Projected zIIP Eligible CPU	CP	IIP	Projected zIIP Eligible CPU
CDC	2401.95	0.00	0.00	0.00	0.00	0.00
DBM1	70.90	44.13	45.93	74.71	45.44	49.00
DDFWORK	2321.77	1422.39	2271.17	2651.80	1265.82	2395.65
DIST	158.21	0.74	0.74	509.98	629.85	972.12
MSTR	92.21	24.41	32.49	59.09	14.35	18.95
PAGENT	0.05	0.00	0.00	0.08	0.00	0.00
TCPIP	58.39	0.00	0.00	44.08	0.00	0.00
Total	5103.52	1491.66	2350.33	3339.74	1955.46	3435.72

Figure 2 CPU Usage: Concurrent Db2 transactions

Here are some key points about Figure 2.

- InfoSphere CDC uses a general-purpose CPU that is called a CP in IBM Z. There is no zIIP eligibility in InfoSphere CDC processing.
- IBM Integrated Synchronization log reading is counted in the Db2 distributed address space, which is zIIP eligible.
- There is not much change from the resource consumption perspective in Pagent Address Space (Policy Agent (PAGENT)) and TCP/IP Address Spaces, as shown in Figure 3 and Figure 4.

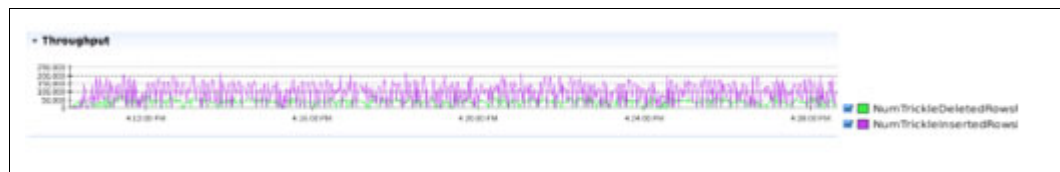


Figure 3 Throughput: Concurrent Db2 transactions

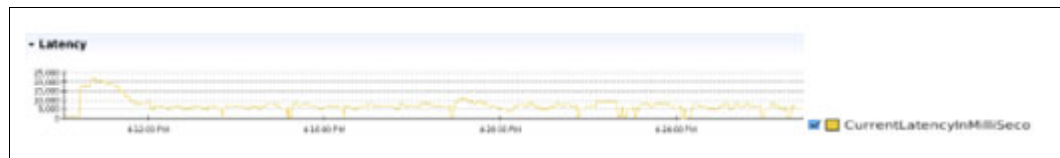


Figure 4 Latency distribution

- There is a concurrent Insert/Delete operation in the source Db2 for z/OS.
- Although the Insert activity was 150,000 - 200,000, the latency was 10 - 20 seconds, which is a significant reduction in replication latency and an improvement in replication throughput compared to InfoSphere CDC, which increased during the ~1-hour test up to 20 minutes. During IBM Integrated Synchronization, the maximum latency during the ~1-hour test was 10 seconds.

Figure 5 shows a CPU usage comparison of the two functions.

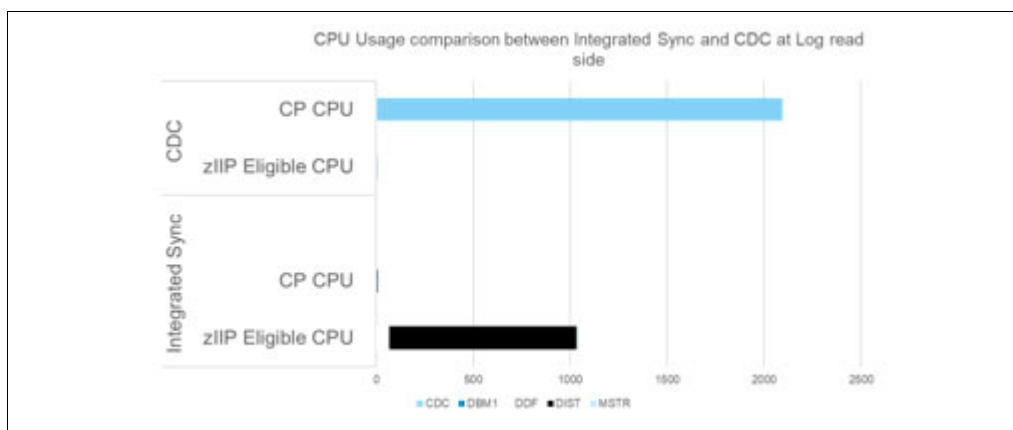


Figure 5 CPU usage comparison between IBM Integrated Synchronization and InfoSphere CDC at the log read side

IBM Integrated Synchronization uses about 49% of CPU time compared to InfoSphere CDC for the same work.

## Comparing InfoSphere Change Data Capture and IBM Integrated Synchronization

The major differences between the two replication technologies are in the following areas:

- ▶ Installation, configuration, and maintenance
- ▶ CPU and latency
- ▶ New functions

IBM Integrated Synchronization is a new Db2 for z/OS replication technology that is streamlined and optimized specifically for Db2 Analytics Accelerator. InfoSphere CDC is a replication product that supports multiple sources and targets. The new log data provider component is integrated into Db2 for z/OS to capture updates to Db2 tables from the log and send the consolidated and encrypted changes to the new log data processor that is integrated into the Accelerator product. No extra product must be installed and maintained.

IBM Integrated Synchronization requires at least Db2 V12 (for more information, see “Prerequisites and recommended maintenance” on page 16), but InfoSphere CDC supports Db2 V11 and Db2 V12. From a maintenance perspective, no extra product must be installed, monitored, and maintained because all updates come through Db2 for z/OS maintenance by applying Db2 PTFs.

The log capture processing is fully zIIP enabled to ensure that there is little impact to the general processing on IBM Z. The streamlined and optimized new design in capture processing with the least possible work left on Db2 for z/OS (only the remote access to the Db2 for z/OS log records) results in less CPU consumption on z/OS compared to InfoSphere CDC.

Figure 6 shows a comparison of the log data provider and log data capture processes.

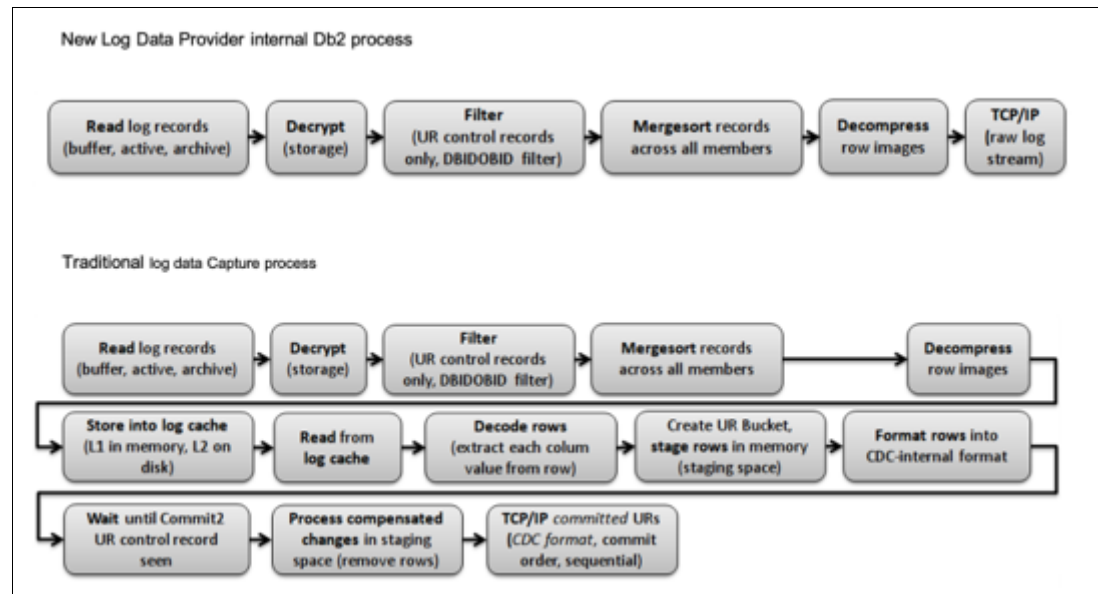


Figure 6 Comparison of log data provider and log data capture processes

In contrast to InfoSphere CDC, the log reader of IBM Integrated Synchronization does not actively process the log, but IBM Integrated Synchronization requests a range of log records. Another major difference of the log reader is that it does not filter log records for committed transactions. All the transaction-related processing, formation of the microbatches, and applying them to the target tables is done on the target side.

Here are some important differences between the two processes:

- ▶ There is no staging space on the source (z/OS).
- ▶ Presumed commit optimization: Uncommitted changes are applied (but not committed) aggressively to achieve low latency for a large transaction.
- ▶ Minimal CPU use on the source:
  - The log streaming interface does minimal work on IBM Z, and can offload to a zIIP.
  - Db2 for z/OS rows flow to the target database. There is no row decoding on the source or the target engine.
  - The capture function does minimal work. It reads raw log records (log buffer, and active or archive log), and decrypts, decompresses, and sends them through TCP/IP.
- ▶ Dynamic apply is based on the number of changed rows for every delete or insert operation, and either a bulk-apply path or a trickle-apply path is used. Bulk-apply is optimized for rows per second throughput, and the trickle-apply path is optimized for the number of changed tables per second throughput.

- Db2 for z/OS log records are parsed on the target. Only entire row images are extracted. Row images are minimally processed (there is no expensive row decoding, and only the key column blobs are extracted).
- Compensated rows are processed much faster. They are batched and merged at End of Unit of Recovery (END UR) time in a single, fast operation.

These improvements result in a significant reduction of replication latency and improve the *true HTAP* query scalability so that using the Wait for Data protocol for concurrent queries has minimal impact on throughput and latency. InfoSphere CDC has a much higher latency that even is higher when you use the Wait for Data protocol.

Functional enhancements, such as the support of non-logged changes to Db2 tables (**LOAD REPLACE** with **DD DUMMY** and **REORG TS** with the **DISCARD** option to delete whole partitions) were implemented only for IBM Integrated Synchronization.

## Different scenarios for data sharing

IBM Integrated Synchronization uses an architecture that closely ties Accelerator to a Db2 for z/OS member for optimal performance. Accelerator with an incremental update enablement establishes a communications session with Db2. Both Accelerator and Db2 keep session-specific states, like the list of objects for incremental updates, the current log position, and a read-ahead buffer for log records on the Db2 side so that optimal throughput can be achieved. These sessions can be long-lived and must be reestablished only when the list of objects changes or when the log reading must be restarted after a communication failure or other problems.

However, this close tie between Accelerator and Db2 requires that the communication session always reaches the same Db2 member when IBM Integrated Synchronization is used in a data sharing environment because the current state in Db2 is not shared between members. This situation is in contrast to the usual communication requirements in data sharing where short-lived requests, usually database queries, can be run on any member for high availability (HA) and load-balancing purposes, and are not bound to a single member for an extended period.

Although this special requirement must be reflected in a dedicated setup, it also offers various scenarios that can be implemented depending on customer preferences and requirements regarding HA.

One common element for the suggested scenarios is that communication between Accelerator and Db2 is done through a dedicated secure port for IBM Integrated Synchronization. Using a dedicated secure port number for IBM Integrated Synchronization provides the needed separation between a regular remote workload coming into Db2 and the log reading requests from IBM Integrated Synchronization. A dedicated secure port can easily be created in Db2 by defining a new location alias, which is described in “Defining a location alias for dedicated secure ports” on page 31.

Using a dedicated secure port for IBM Integrated Synchronization has the following advantages over using an existing secure port:

- Can be set up on each member individually.
- Incoming requests for IBM Integrated Synchronization can be allowed or disallowed by starting or stopping the location alias, for example, for load balancing or when maintenance must be done on a member.

- Network rules for IBM Integrated Synchronization can be different from general-purpose rules, for example:
  - Specific encryption settings by using a different server certificate
  - Limiting incoming requests to specific IP addresses or ranges
  - Different distribution methods for distributed dynamic VIPAs (DDVIPAs)

## Scenario 1: Dedicated single member for IBM Integrated Synchronization

The simplest scenario is to dedicate a single member as the endpoint for IBM Integrated Synchronization connections. The location alias for IBM Integrated Synchronization is added only on a single member, and all communications must use this member. If only one member has the dedicated secure port defined, either the direct member IP address or the DDVIPA IP address can be used to enable incremental updates. Using a DDVIPA IP address over the direct member IP address has the advantage that the IP address does not have to be changed on the Accelerator in case the dedicated member is down and a different member must be dedicated to accept connections.

Figure 7 shows two LPARs with their IP addresses (10.7.1.8 and 10.7.1.9). Each logical partition (LPAR) hosts one member (DB2A and DB2B) of the Db2 data sharing group (DB2G). In addition, a DDVIPA with VIPA 10.7.1.99 is defined to distribute incoming connections to both LPARs.

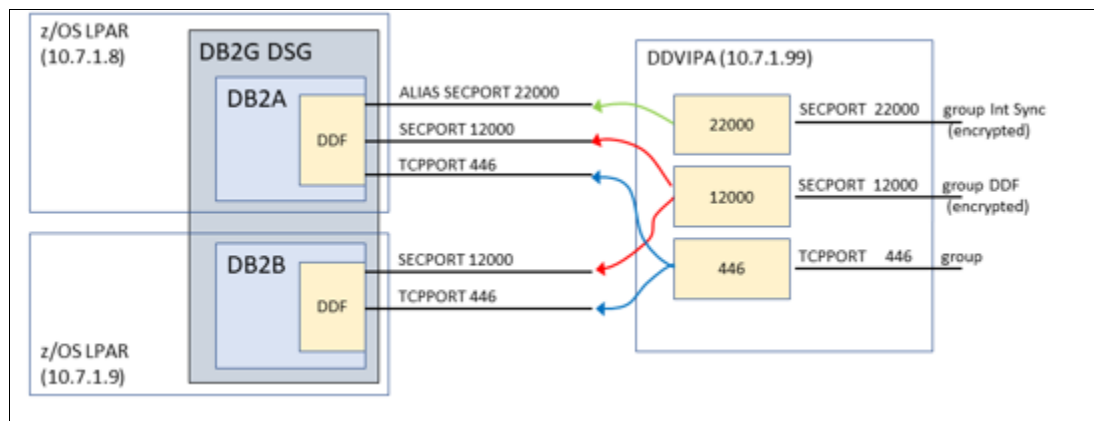


Figure 7 Dedicated single data sharing member for IBM Integrated Synchronization

For a regular remote workload, both members have the regular TCPPOINT (446) and the SECPOINT (12000) defined. A regular SQL workload can either connect directly to a member by using the LPAR IP address or by using the DDVIPA address. Requests for both ports are directed to any of the two (or further) members, depending on the VIPADISTRIBUTE definition.

Requests for IBM Integrated Synchronization use the dedicated secure port number 22000, and because it is defined only on member DB2A, connections are possible by using either the LPAR address of 10.7.1.8 or (a best practice) the DDVIPA address of 10.7.1.99. Even when connecting through the DDVIPA, no connection can be established to member DB2B because the dedicated secure port is not defined and started on that member.

This scenario is simple and has a significant disadvantage: Only one member handles the workload for IBM Integrated Synchronization, and if that member has either planned or unplanned downtime, no incremental updates can be performed. Therefore, this solution is not recommended except as a sandbox system so that you can become familiar with this technology.

## Scenario 2: Manually controlled dedicated member for IBM Integrated Synchronization

Evolving from “Scenario 1: Dedicated single member for IBM Integrated Synchronization” on page 11, this scenario is where the administrator controls which Db2 member is running the IBM Integrated Synchronization workload. The solution still has only a single member that runs the IBM Integrated Synchronization workload, but you can bring down a Db2 member while minimally impacting incremental updates. It can be used to react to unforeseen outages by manually bringing up IBM Integrated Synchronization on a different member to provide basic HA capability.

Figure 8 shows a multiple manually controlled dedicated members scenario.

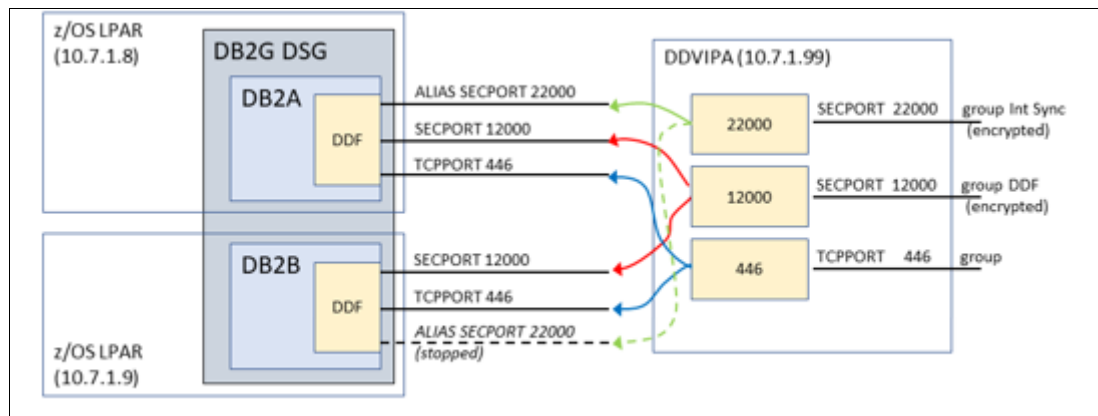


Figure 8 Multiple manually controlled dedicated members

For this setup, the location alias for IBM Integrated Synchronization that uses the dedicated secure port is added to both (or more) members of the data sharing group. However, the Db2 administrator ensures that only one alias is started at a time.

Incremental updates on Accelerator are set up always to use the DDVIPA address and the dedicated secure port of the location aliases. If only a single member is listening to the dedicated secure port, DDVIPA cannot distribute incoming requests to multiple members, and the dedicated connection requirement is fulfilled.

When the member that is running incremental update workload must be brought down for any reason, the location alias on that member must be stopped, which prevents any further requests from coming in to that member. Concurrently or soon afterward, the location alias on the replacement member should be started to allow continued incremental update operations.

When stopping the location alias on the first member, Accelerator detects a connection loss and retries the connection for a short period. Should the secure port on the replacement member become available within the retry period, then Accelerator establishes a new connection through the DDVIPA address. A new session with the replacement member is started when Accelerator cannot reuse the previous session identifier. During that failover processing, Accelerator issued several messages indicating the loss of communication, the failure of the previously used session, and the establishment of the new session. If the replacement member does not come up within the retry period, then replication stops on Accelerator, and then can be restarted when the target Db2 member is available again.

For a more controlled move to a different Db2 member to avoid error messages in the replication events and on the z/OS SYSLOG, complete the following steps:

1. Stop replication on Accelerator.
2. Stop the location alias on the active Db2 member.
3. Start the location alias on the replacement member.
4. Start replication on Accelerator.

Because Accelerator can restart IBM Integrated Synchronization sessions automatically, the impact of the move to a different Db2 member is relatively small. There is an impact that is associated with each switch to a different Db2 member because log records that were read by the original Db2 member are thrown away instead of being collected from the read ahead buffer, and a new session must be established on the replacement member, which has some initial cost that is associated too.

**Note:** Although the session restart is done automatically, it still requires a dedicated connection to a single Db2 member after the restart. Should more than one location alias start concurrently, it is likely that the DDVIPA distributes each connection request from Accelerator to a different member, which starts a series of session restarts. This situation makes incremental updates ineffective if not unusable, and should be avoided.

By using this setup with manually controlled Db2 members, it is easy to control which Db2 member must take the CPU load that is needed to read the log records that are needed for incremental updates, allowing for manual workload balancing across different members if another Db2 workload already is running on dedicated members. This setup also provides basic failover capability to react to any planned or unplanned outages.

### Scenario 3: Automatically controlled dedicated member for IBM Integrated Synchronization

Although the previous scenarios do not use the DDVIPA function to its full extent, but use the DDVIPA to provide only a single common IP address for incoming requests to Db2, it is possible to run with an automated setup that also provides HA.

As described in “Scenario 2: Manually controlled dedicated member for IBM Integrated Synchronization” on page 12, Accelerator allows automatic restart of incremental update sessions if the previous session is no longer reachable for any reason. Using a special DDVIPA setup, this situation can be automated without requiring any manual intervention.

Usually, DDVIPA is used to distribute evenly the incoming requests among all available Db2 members of a data sharing group. However, this distribution method does not work with IBM Integrated Synchronization requests because this technology requires persistent connections between a specific instance of Accelerator and a Db2 member.

The configuration for DDVIPA distribution options is done on a port level, which means that for every port that is used on a z/OS system, it is possible to use a different distribution logic. This configuration also implies that for a HA setup for IBM Integrated Synchronization, it is mandatory to use a Db2 location alias with a dedicated secure port to separate the distribution method from the one that is used for a regular SQL workload.

Figure 9 illustrates the full DDVIPA setup.

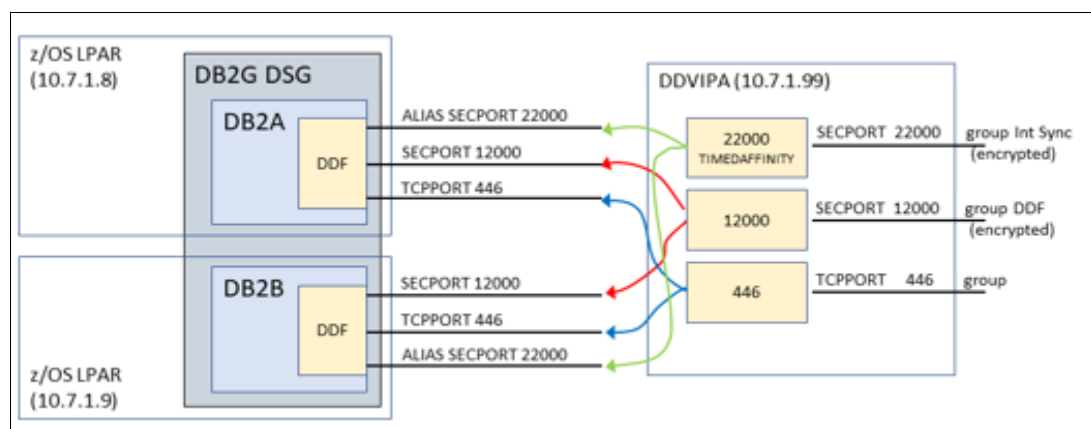


Figure 9 Automatically controlled members with DDVIPA for high availability

At first glance, the setup looks almost identical to the manually controlled setup. However, there are some important differences:

- It is possible to start the location alias on two or even more Db2 members that can be selected as IBM Integrated Synchronization targets.
- The DDVIPA must have the special option TIMEDAFFINITY specified on the distribution statement for the dedicated secure port.

As the name of this option implies, it specifies a period for which incoming connection requests from the same source (determined by IP address) are directed to the same target and do not go through a new distribution logic. With this option, any established connection remains connected to the same Db2 member if TIMEDAFFINITY specifies a higher value than the usual time between requests.

IBM Integrated Synchronization has a timeout value of 60 seconds for the Db2 log reader tasks. If no new request comes in within 60 seconds for each active session, the respective session is terminated automatically by Db2, and the resources are cleaned up. Accelerators make sure to send new requests within this period to keep the log reader task active. A TIMEDAFFINITY value of 60 (specified in seconds) lets the DDVIPA distribute all repeating requests to the same member if both sides are running.

When Db2 is no longer available to receive the request (no process is listening to the port anymore), DDVIPA ignores the TIMEDAFFINITY parameter and starts a new distribution process to find the next available Db2 member that can accept a connection at the specified port address. After the new connection is established, Accelerator remains connected to that member until Accelerator cannot reach that member.

When Accelerator stops its requests to Db2 and restarts again after the 60 seconds have expired, it is likely that Accelerator will be distributed to a different Db2 member than before. Usually, this situation occurs only if incremental updates were suspended, so a new session must be created.

This DDVIPA setup also supports multiple instances of Accelerator that use IBM Integrated Synchronization. Each instance of Accelerator can be distributed to a different Db2 member or the same Db2 member depending on the other distribution options, and they remain connected to their assigned Db2 members independently.



In this scenario, failover to a different Db2 member is performed automatically. The administrator sees events like intermediate error messages when a new session must be established, but otherwise incremental updates remain active.

Planned maintenance can easily be performed in this setup by simply stopping the location alias on the member that must be brought down. While one or more other Db2 members are still available with a started location alias for IBM Integrated Synchronization, the incoming connections are directed to those other members, and the Db2 member is ready to be brought down.

## Scenario 4: High availability setup for IBM Integrated Synchronization with a directed workload

In larger data sharing groups, it might a best practice to have a HA setup that provides automatic failover to a different member when one member fails. However, if some members of the data sharing group should be spared from getting an IBM Integrated Synchronization workload, they cannot start or create the location alias. For example, an 8-way data sharing group might use only two members as endpoints for IBM Integrated Synchronization workload, and the remaining six members process the remaining workload, as shown in Figure 10.

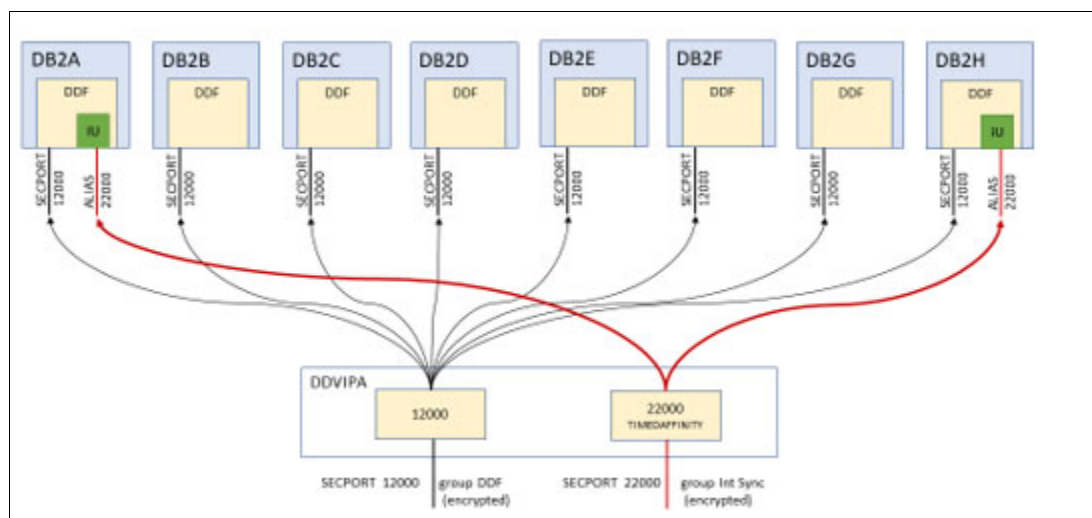


Figure 10 High availability setup with dedicated members for dedicated synchronization

# Setup details, prerequisites and recommended maintenance, and installation and configuration

The setup for IBM Integrated Synchronization can be divided into two major areas that involve several steps each:

- ▶ Setting up the z/OS environment:
  - Security setup in RACF (the certificates that are needed for encryption, a dedicated user ID, an access profile, and data set access).
  - Security setup for the network (AT-TLS configuration).
  - Network setup for data sharing groups.
  - Db2 setup (correct maintenance, encrypted access, and privileges for the incremental update user).
- ▶ Activating incremental updates on Accelerator:
  - Transfer the certificate for the verification of an encrypted connection to Db2.
  - Enable incremental updates in the configuration console.
  - Enable replication for Accelerator.
  - Enable replication for the table.

These setup steps require the cooperation of multiple system administration disciplines, like z/OS system programmers, RACF security administrators, z/OS network administrators, and Db2 system programmers and administrators.

## Prerequisites and recommended maintenance

IBM Integrated Synchronization reduces the setup and maintenance efforts that are needed to replicate tables between Db2 for z/OS and IBM Db2 Analytics Accelerator. Because the capture function is integrated into the Db2 for z/OS engine, the need to install, configure, and maintain a separate replication product, even if it is already distributed with IBM Db2 Analytics Accelerator, is obsolete.

All required maintenance for Db2 Analytics Accelerator is described at [Prerequisites and Maintenance for IBM Db2 Analytics Accelerator for z/OS V7](#).

IBM Integrated Synchronization has the following prerequisites:

- ▶ Db2 12 for z/OS with IBM Integrated Synchronization enabling APAR PH06628 (PTF UI63356) installed, running at function level V12R1M500.
- ▶ In addition to the enablement APAR, the following APARs (PTFs) are mandatory for correct function:
  - APAR PH23895 - PTF UI69536: Improve termination processing for log reader tasks, which prevents hanging distributed connections.
  - APAR PH28849 - PTF UI71876: The storage usage in the DBM1 address space increases continuously, mainly in the Compression Dictionary according statistics trace.

- ▶ Chronological list of recommended APARs (PTFs):
  - APAR PH19181 - PTF UI66752: ABEND04E RC00D34454 in DSNLXLLM.DSNLJXUS:0002 when doing many loads in parallel.
  - APAR PH19886 - PTF UI67234: ABEND04E RC00C90050 in DSNILGRT+01458 when loading tables or disabling replication, or LSN moving backwards error.
  - APAR PH20587 - PTF UI67915: ABEND DC2-5C004221 when allocating buffer for log records. (This error is not specific to IBM Integrated Synchronization.)
  - APAR PH21187 - PTF UI67814: Privilege check fails with ABEND04E RC00E70005 AT DSNXACAE OFFSET00BCE when setting up IBM Integrated Synchronization while an authorization exist is active.
  - APAR PH21419 - PTF UI68477: Fix error handling for memory allocation failures.
  - APAR PH18334 - PTF UI70139 and APAR PH26681 - PTF UI71042: Provide statistics and accounting enhancements for IBM Integrated Synchronization incremental update method.
  - APAR PH26450 - PTF UI70986: ABEND 0C4-00000011 in DSNVDTA:05C0C or ABEND 0C1-00000001 in DSNAPRHX:007E when running incremental updates with IBM Integrated Synchronization.
  - APAR PH27992 - PTF UI71324: Retry logic for temporary decompression errors to avoid setting tables in error and having to reload tables.
  - APAR PH29443 - PTF UI71815: Client unable to activate IBM Integrated Synchronization for Db2 tables with a decimal column that was created before Db2 V7.
  - APAR PH30312 - PTF UI72842: Serviceability enhancement.
  - APAR PH30397 - PTF UI72591: Incorrect latency displayed if log reading is slow.
  - APAR PH31772 - PTF UI73158: Missing table schema details for function in IDAA V7.5.4.
  - APAR PH32510 - PTF (open): Incorrect zIIP eligible times reported in IFCID 2 fields QISTLRZE and Q8STLRZE for Integrated Synchronization log reader tasks.
- ▶ To determine the list of recommended Db2 PTFs, run the SMP/E **REPORT MISSINGFIX** command on your z/OS systems and specify Fix Category:
 

```
FIXCAT(IBM.DB2.AnalyticsAccelerator.V7R1,IBM.DB2.AnalyticsAccelerator.V7R5)
```
- ▶ Installing the PTFs for APARs PH20587 (general Db2), PH30312, and PH31772 pull in all required maintenance.
- ▶ IBM Db2 Analytics Accelerator for z/OS Version 7.5.0 or later.
- ▶ The following z/OS and TCP/IP prerequisites must be installed and active to use AT-TLS:
  - IP ICSF (IBM Encryption Facility for z/OS).
  - To activate AT-TLS, the **TTLS** parameter must be added to the **TCPCONFIG** profile configuration statement of TCP/IP.
  - PAGENT needs to be started.

If these components are not installed or set up, see the product documentation and the following references:

  - *IBM z/OS V2R2 Communications Server TCP/IP Implementation: Volume 4 Security and Policy-Based Networking*, SG24-8363, found at: <http://www.redbooks.ibm.com/abstracts/sg248363.html?Open>

- *DB2 for z/OS: Configuring TLS/SSL for Secure Client/Server Communications*, REDP-4799, found at:

<http://www.redbooks.ibm.com/redpapers/pdfs/redp4799.pdf>

- DDF with a secure port that is configured for network encryption through AT-TLS.
- If a private network is used between Db2 for z/OS and IBM Db2 Analytics Accelerator, then DDF must listen on the private network and to the public network address. This configuration is achieved by using the INADDR\_ANY setup. If DDVIPA is used with a private network, extra definitions must be made for the private network addresses.
- It is a best practice to have a dedicated SECPORT instead of using the standard Db2 SECPORT.

## Security setup in RACF

When Accelerator uses IBM Integrated Synchronization, it must be authenticated to use the Db2 for z/OS interface to read from the log. This action requires a user ID with specific privileges, so as a best practice use a dedicated user ID for that purpose. This user ID must be able to authenticate by using a password or PassTicket.

### Optional: Using PassTickets with Db2

The RACF PassTicket class must be activated by running a TSO session command:

```
SETROPTS CLASSACT(PTKTDATA) RACLIST(PTKTDATA)
SETROPTS GENERIC(PTKTDATA)
```

The RACF profile for the application (Db2) is defined by running these TSO session commands:

```
RDEFINE PTKTDATA <appName> SSIGNON(KEYMASKED(<key>)) -
  APPLDATA('NO REPLAY PROTECTION')
```

- <appName> is the name of the application that requests and uses the PassTickets, which in this case is the Db2 subsystem. If necessary, run the Db2 **-DISPLAY DDF** command to discover the application name as the value for <appName>:
  - If GENERICLU is defined, use the second part of GENERICLU for <appName>.
  - If GENERICLU is not defined, use the second part of LU-name for <appName>.
  - If neither GENERICLU or LU-name are defined, use the value of the IPNAME for <appName>.
- <key> is a session key with a value of 16 hexadecimal digits (for an 8-byte or 64-bit key). The session key must be identical to the key in the PassTicket definition in each RACF instance. The key for each application must be the same on all systems in the configuration. **APPLDATA('NO REPLAY PROTECTION')** is the option that you can use to permit reuse of the same PassTicket multiple times for a period of up to 10 minutes. This option is required when using IBM Integrated Synchronization because multiple requests might be sent to Db2 within a second before a new PassTicket is created.

**Note:** A PassTicket is generated based on the current time in seconds. If multiple requests must generate a PassTicket, they get the same value if all of them are generated within the span of the same second. If **'NO REPLAY PROTECTION'** is not used, a second generated PassTicket is not allowed, which causes the process to fail.

- The dash (-) at the end of the example is the line continuation character, and it must be omitted if the entire command fits on a single line.

The following TSO commands define RACF profiles for PassTicket generation:

```
RDEFINE PTKTDATA IRRPTAUTH.<appName>.* UACC(NONE)
PERMIT IRRPTAUTH.<appName>.* ID(<db2-ddf-user>) ACCESS(UPDATE) -
CLASS(PTKTDATA)
```

<db2-ddf-user> is the user ID under which the Db2 DDF address space (<ssid>DIST) runs.

The PTKTDATA class is refreshed and activated by running the following command:

```
SETOPTS RACLIST(PTKTDATA) REFRESH
```

When incremental updates are enabled later in the configuration console, the application name <appName> and the sign-on key <key> are needed, so that information must be passed on to the Accelerator administrator.

## Permitting remote connections to access the Db2 interface to read log records

After the user ID is created, it must be given access to special profiles in the DSNR general resource class in RACF. These RACF profiles are selected to allow connections to DDF in general and use the IBM Integrated Synchronization API in particular. The names of the profiles consist of the subsystem name followed by a dot and the constants DIST and ACCEL respectively. The owner of these profiles should be the user ID under which the Db2 DDF address space runs. The IBM Integrated Synchronization user ID needs READ access to both of these profiles.

To create the RACF profiles and permit access to them, run the following TSO session commands:

```
RDEFINE DSNR (<ssid>.DIST) OWNER(<db2-ddf-user>) UACC(NONE)
PERMIT <ssid>.DIST CLASS(DSNR) ID(<int-synch-user>) ACCESS(READ)
RDEFINE DSNR (<ssid>.ACCEL) OWNER(<db2-ddf-user>) UACC(NONE)
PERMIT <ssid>.ACCEL CLASS(DSNR) ID(<int-synch-user>) ACCESS(READ)
```

- <ssid> is the 4-character Db2 subsystem name.
- <db2-ddf-user> is the user ID under which the Db2 DDF address space (<ssid>DIST) runs.
- <int-synch-user> is the user ID that is created for incremental updates by using IBM Integrated Synchronization, for example, DB2SYNC.

## Granting access to data sets that are accessed by stored procedures

When IBM Integrated Synchronization detects an error condition for a table, that table is removed from acceleration to prevent Db2 from returning stale data because no more updates are synchronized to Accelerator. The incremental update process invokes the stored procedure SYSPROC.ACCEL\_SET\_TABLES\_ACCELERATION to disable acceleration. Accelerator stored procedures access a few configuration data sets during run time, so the user ID that is used for IBM Integrated Synchronization also must be granted access to those configuration data sets so that the stored procedure can be invoked correctly.

The following TSO session commands are needed to permit this data set access:

```
PERMIT '<aqtENV-data-set>' ID(<int-synch-user>) ACCESS(READ)
PERMIT '<aqtDEFTR-data-set>' ID(<int-synch-user>) ACCESS(READ)
```

- ▶ <aqtENV-data-set> is the data set name that contains the AQTENV file in the started task procedure of the Workload Manager (WLM) environment.
- ▶ <aqtDEFTR-data-by set> is the data set name that contains the AQTDEFTR file in the started task procedure of the WLM environment.
- ▶ <int-synch-user> is the user ID that is created for incremental updates using IBM Integrated Synchronization.

In addition to the data set permissions, the <int-synch-user> user ID must have an OMVS segment defined for it in RACF.

## Creating certificates that are needed for SSL encryption

IBM Integrated Synchronization requires SSL encryption between Db2 and Accelerator to protect the sensitive data being transferred over the network. SSL encryption uses a private/public key-based encryption method where the private and public keys are stored in certificates.

When a connection to Db2 requests encryption, the Db2 side uses the private key of the certificate to encrypt the data and the client (in this case Accelerator) uses the public key to decrypt the data.

More trust in the encrypted connection is provided when the Db2 certificate is signed by a certificate authority (CA). The CA is identified by a certificate and public key, which must be known by Accelerator to trust the connection to Db2.

On z/OS, certificates are created and stored in RACF. When creating a certificate, the administrator must assign a unique label to it so that it can be used in any subsequent RACF commands to refer to each individual certificate. To access a certificate from an application, it is assigned to a *key ring*, which is simply a container for one or more certificates.

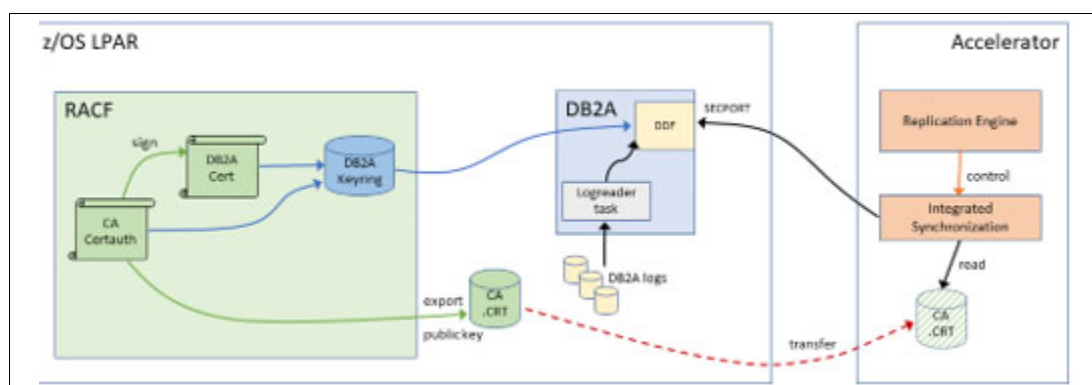


Figure 11 RACF: Certificates and key rings for IBM Integrated Synchronization

The easiest way to handle certificates is to use *self-signed certificates*. These certificates are signed by a private CA instead of a commonly known CA like VeriSign or the like. Large corporations often have their own CA established with the infrastructure for issuing and renewing certificates, and the trust infrastructure.

Usually, IBM Integrated Synchronization communicates on a private network between Accelerator and Db2. Therefore, it is sufficient to use a self-signed certificate instead of an official certificate from a third-party CA or a company's internal CA. However, company policy might require using those external or internal CAs for any encryption purpose.

## Defining RACF facilities for certificate handling

If not already done, you must use RACF to set up the facilities to handle certificates. While doing so, access to those facilities must be granted to the user ID that is used to run the Db2 DDF address space. These actions can be done by using these TSO session commands:

```
SETROPTS CLASSACT(DIGTCERT DIGTRING)
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(NONE)
RDEFINE FACILITY IRR.DIGTCERT.LIST UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) -
    ID(<db2-ddf-user>) ACCESS(CONTROL)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) -
    ID(<db2-ddf-user>) ACCESS(READ)
SETR RACLIST (DIGTRING) REFRESH
SETR RACLIST (DIGTCERT) REFRESH
SETR RACLIST (FACILITY) REFRESH
```

<db2-ddf-user> is the user ID under which the Db2 DDF address space (<ssid>DIST) runs.

## Creating a self-signed certificate

The CA certificate, or signer certificate, or as RACF describes it, the CERTAUTH certificate, is needed when no other CA is available to create and sign the certificate to be used by Db2 for encryption. The main information that is needed for the signer certificate is the subject of the certificate, which is plain text describing the certificate. Various fields can be used to describe any certificate, but it is sufficient to specify only the canonical name (CN) for the subject.

The following commands are used to create a CA certificate:

```
RACDCERT CERTAUTH GENCERT KEYUSAGE(CERTSIGN) -
    SUBJECTSDN(CN('<ca-certificate-subject>')) -
    NOTAFTER(DATE(<valid-until-date>)) -
    WITHLABEL('<ca-certificate-label>')
```

- ▶ '<ca-certificate-subject>' is a descriptive name of the CA certificate to be created, for example, 'My Org CA'.
- ▶ <valid-until-date> is the date that the certificate is valid before it must be renewed. The valid format is YYYY-MM-DD.
- ▶ '<ca-certificate-label>' is the unique label that identifies this certificate within RACF.

After the CA certificate is created, the public key that identifies this certificate can be exported to a data set on z/OS because it is required later when setting up incremental updates on Accelerator. Use the correct format during the export because only the DER format is recognized by Accelerator. Because the DER format is a binary format, the binary structure must be preserved during data set handling.

To export the CA certificate, run the following command:

```
RACDCERT CERTAUTH EXPORT(LABEL('<ca-certificate-label>')) -  
      DSN('<ca-certificate-dsn>') -  
      FORMAT(CERTDER)
```

- ▶ '<ca-certificate-label>' is the unique label that references the CA certificate within RACF.
- ▶ '<ca-certificate-dsn>' is the data set name to which the public key is exported. The data set name should use a .crt extension to identify it as a DER formatted certificate.

The exported CA certificate is transferred to Accelerator during a later installation step. For this transfer, the certificate either must be copied to the HFS path that is specified by variable *AQT\_HOST\_PACKAGE\_DIRECTORY* in the AQTENV environment settings, or it must be transferred to a workstation where IBM Db2 Analytics Accelerator Studio can be run. In both cases, the certificate must be transferred in binary, and the target file should use a lowercase extension of .crt.

To transfer the certificate to the HFS path that is specified by the variable *AQT\_HOST\_PACKAGE\_DIRECTORY* in the AQTENV environment settings, run the following TSO:

```
OPUT '<ca-certificate-dsn>' '<ca-certificate-hfs-pathname>' binary
```

- ▶ '<ca-certificate-dsn>' is the data set name to which the public key of the CA certificate was exported.
- ▶ '<ca-certificate-hfs-pathname>' is the target file name of the exported CA certificate in HFS where the path is specified in the AQTENV variable *AQT\_HOST\_PACKAGE\_DIRECTORY*. The data set name should use an extension of .crt (all in lowercase) to identify it as a DER formatted certificate.

Alternatively, the exported CA certificate can be transferred to a workstation running IBM Db2 Analytics Accelerator Studio in binary format by using existing file transfer mechanisms like FTP or file transfer in the emulator.

## Using an existing CA certificate and infrastructure

If you want to use an existing CA infrastructure, then you can skip the creation of the CA certificate. Exporting the existing CA certificate from RACF is still required for the setup of incremental updates on Accelerator, and the <ca-certificate-label> must identify the CA certificate that exists in RACF.

## Creating a self-signed server certificate for Db2

The Db2 server certificate identifies Db2 as the server and provides the private and public key for encryption. It must be signed by the previously created or referenced CA certificate.

Normally, a server certificate is used to make sure that a connection is reaching the exact server that was intended. For this verification, the subject of the certificate must reflect the exact server address, usually a fully qualified hostname (including the domain, which is also used to connect to a server).

Because it is a best practice to use a private network for Accelerator to Db2 communication where domain name services are usually not available and the addressing is done based on IP addresses, the Db2 certificate does not need to contain the private IP address. Instead, use the public fully qualified hostname for the CN in the subject of the certificate. Accelerator does not require that the IP address match the CN, but other users of the public company network can use the same certificate for encrypted connections with hostname verification turned on (which is the default for most clients).



The following commands are used to create and sign the Db2 server certificate:

```
RACDCERT ID(<db2-ddf-user>) GENCERT -  
          SUBJECTSDN(CN('<db2-certificate-subject>')) -  
          NOTAFTER(DATE(<valid-until-date>)) -  
          WITHLABEL('<db2-certificate-label>') -  
          SIGNWITH(CERTAUTH LABEL('<ca-certificate-label>'))
```

- ▶ <db2-ddf-user> is the user ID under which the Db2 DDF address space (<ssid>DIST) runs. By specifying the ID, the certificate is owned by the Db2 DDF address space user and no additional access permissions need to be granted. This option can be omitted if access is granted by specific RACF PERMIT statements.
- ▶ <db2-certificate-subject> is a fully qualified hostname to connect to Db2 or the Db2 data sharing group, for example, 'mydb2group.company.com'.
- ▶ <valid-until-date> is the date after which the certificate is no longer valid. The valid format is YYYY-MM-DD.
- ▶ <db2-certificate-label> is the unique label that identifies this Db2 server certificate within RACF.
- ▶ <ca-certificate-label> is the unique label that references the CA certificate within RACF.

## Creating a key ring and store certificates

For encrypted connections to Db2, the certificates must be accessible from a key ring in RACF. Both the CA certificate and the Db2 server certificate must be stored in that key ring. No other certificate should be added to the key ring because the default certificate from the key ring is used for encryption when following the setup instructions in this paper.

In data sharing environments, it is possible to use a single key ring in RACF for all members if the Db2 server certificate was created with the group hostname and all Db2 members use the same user ID under which the DDF address spaces run. If those user IDs do not match, they must be given access to the key ring, which is described in the documentation for RACF.

The following commands create the key ring that is owned by the Db2 DDF user ID and assign the previously created certificates to it:

```
RACDCERT ID(<db2-ddf-user>) ADDRING(<db2-key-ring>)  
RACDCERT ID(<db2-ddf-user>) CONNECT(CERTAUTH -  
          LABEL('<ca-certificate-label>') -  
          RING(<db2-key-ring>))  
RACDCERT ID(<db2-ddf-user>) CONNECT(ID(<db2-ddf-user>) -  
          LABEL('<db2-certificate-label>') -  
          RING(<db2-key-ring>) DEFAULT)
```

- ▶ <db2-ddf-user> is the user ID under which the Db2 DDF address space (<ssid>DIST) runs. By specifying the ID, the key ring is owned by the Db2 DDF address space user and no extra access permissions must be granted. This option can be omitted if access is granted by other RACF means.
- ▶ <db2-key-ring> is the unique identifier of the key ring to be used by Db2 encryption. It should not contain blanks or any other special characters.
- ▶ <ca-certificate-label> is the unique label that references the CA certificate within RACF.
- ▶ <db2-certificate-label> is the unique label that references the Db2 server certificate within RACF.

## Activating the certificate changes in RACF

To activate the certificates and the key ring, run the following commands:

```
SETR RACLIST (DIGTRING) REFRESH
SETR RACLIST (DIGTCERT) REFRESH
SETR RACLIST (FACILITY) REFRESH
```

## Security setup for network

This section describes the security setup for the network.

### AT-TLS setup

All communication between Accelerator and Db2 for incremental updates must be encrypted to protect the data that is contained in the log records. Db2 does not implement the encryption by itself, but uses the Application Transparent Transport Layer Security (AT-TLS) support that is a component of z/OS Communications Server. AT-TLS provides a communication layer that transparently encrypts connections coming in to Db2. It does all the encryption handshaking and processing on behalf of Db2.

AT-TLS must be set up to identify the connections going into Db2 that must be encrypted and where the certificates for the encryption can be found. A full description of AT-TLS and the configuration options can be found at [IBM Knowledge Center](#).

An important z/OS system component that is required for AT-TLS to run is PAGENT, which is part of z/OS Communications Server. PAGENT handles all the network policies, including encryption policies from AT-TLS. A full description of PAGENT and the configuration options can be found at [IBM Knowledge Center](#).

For this paper, assume that PAGENT is set up and running and that only AT-TLS settings must be updated to enable encryption to Db2.

### Verifying that AT-TLS is enabled

AT-TLS functions are enabled in the TCP/IP profile. The profile is referenced in the Job Control Language (JCL) for the TCPIP started task in the PROFILE DD card. To enable AT-TLS, the profile must have the TTLS entry on the **TCPCONFIG** statement. If the statement is missing, it must be added while leaving all other options unchanged, for example:

```
TCPCONFIG
TTLS      ; Enable AT-TLS function
...
```

### Identifying or creating the AT-TLS configuration data set

After AT-TLS is enabled in TCP/IP, the configuration data set must be specified in the PAGENT parameters. The parameters are referenced in the JCL for the PAGENT started task in the **PARM** option of the **EXEC** statement. You can either specify the parameter file directly or reference a DD card that points to the parameter file.

Then, the PAGENT parameter file must be updated to contain a reference to the AT-TLS configuration, which is referred to by the **TTLSSConfig** option, for example:

```
TTLSSConfig //'SYS1.TCPPARMS(TTLS)'
```

The AT-TLS configuration contains statements for all connections that must be encrypted by this service. The starting point for any new connection that must be added is **TTLRule**. Each **TTLRule** further specifies characteristics that identify the connections, for example, IP addresses, port numbers, and started task names.

The AT-TLS configuration uses a hierarchical format where each option is identified by a unique name and can be referenced in other options to share common definitions across the environment. New definitions can simply be added to the end of the configuration file if the new identifiers are unique.

## Shared AT-TLS options for multiple Db2 subsystems

While planning ahead, specify some of the AT-TLS options in a way that allows them to be reused by all Db2 subsystems in an environment because they usually do not change. These options include the basic enablement of encryption and the ciphers that are used to encrypt the connections.

Encryption in general is enabled by the group-action option by using the following statement:

```
TTLGroupAction <db2-group-action-name>
{
    TTLEnabled On
}
```

<db2-group-action-name> is a unique identifier that can be referenced in the AT-TLS configuration. It should not contain blanks or special characters for easier reference.

Similarly, the **TTLSCipherParms** contains a list of ciphers to be used for encryption. The following example mainly uses Galois/Counter Mode (GCM) ciphers, which show the best performance results in the lab environments where IBM Integrated Synchronization was developed and tested because the ciphers provide hardware support on both Accelerator and z/OS. Basically, all ciphers that are supported on both platforms can be used, but CPU performance and network throughput can vary when selecting different ciphers.

To define a basic list of ciphers to be used by encryption to Db2, use the following statements:

```
TTLSCipherParms <db2-cipher-parms-name>
{
    V3CipherSuites    TLS_RSA_WITH_AES_128_GCM_SHA256
    V3CipherSuites    TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
    V3CipherSuites    TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
    V3CipherSuites    TLS_RSA_WITH_AES_128_CBC_SHA
}
```

<db2-cipher-parms-name> is a unique identifier that can be referenced in the AT-TLS configuration. It should not contain blanks or special characters for easier reference.

## Db2 subsystem-specific AT-TLS options

Settings for each Db2 subsystem start with a **TTLRule** statement and identify the specific parameters that identify the connection, as shown in the following statements:

```
TTLRule <db2-subsystem-rule-name>
{
    LocalPortRange    <db2-subsystem-secport>
    JobName            <db2-subsystem-ddf-task>
    Direction          Inbound
    TTLGroupActionRef  <db2-group-action-name>
    TTLEnvironmentActionRef <db2-subsystem-environment-name>
}
```

- ▶ `<db2-subsystem-rule-name>` is a unique identifier for the Db2 subsystem-specific rule in the AT-TLS configuration. It should not contain blanks or special characters for easier reference.
- ▶ `<db2-subsystem-secport>` is the secure port number for Db2 that should be encrypted. This option can specify only a single port number. Encryption of multiple ports is described in “Enabling multiple secure ports for a Db2 subsystem” on page 26.
- ▶ `<db2-subsystem-ddf-task>` is the name of the Db2 subsystem's DDF address space `<ssid>DIST`.
- ▶ `<db2-group-action-name>` is the unique identifier that references the shared definition in the AT-TLS configuration to enable encryption.
- ▶ `<db2-subsystem-environment-name>` is the unique identifier that references the Db2 subsystem-specific environment action in the AT-TLS configuration.

The environment action that is referenced in the last option is used to specify the RACF key ring containing the certificates and the cipher parameters by using the following statement:

```
TTLSEnvironmentAction <db2-subsystem-environment-name>
{
    TTLSKeyRingParms
    {
        Keyring          <db2-key-ring>
    }
    HandShakeRole        Server
    TTLSCipherParmsRef   <db2-cipher-parms-name>
}
```

- ▶ `<db2-subsystem-environment-name>` is a unique identifier that can be referenced in the AT-TLS configuration. It should not contain blanks or special characters for easier reference.
- ▶ `<db2-key-ring>` is the unique identifier of the key ring in RACF that is used by this Db2 subsystem.
- ▶ `<db2-cipher-parms-name>` is the unique identifier that references the shared cipher parameters in the AT-TLS configuration.

## Enabling multiple secure ports for a Db2 subsystem

Because it is a best practice to use a dedicated secure port for IBM Integrated Synchronization that is different from the default secure port of Db2, the **TTLRule** for a Db2 subsystem must be able to define more than one port number for which the encryption details should be valid if the same encryption settings are used for both ports. If different encryption settings are used, then the previous definitions need to be duplicated only for each secure port while keeping the identifiers unique.

Instead of defining a single `LocalPortRange` in the example in “Db2 subsystem-specific AT-TLS options” on page 25, multiple ports are put into a `LocalPortGroup`, which is usually defined as a separate statement and then referred to in the main **TTLRule**, as outlined in the following statements:

```
TTLRule <db2-subsystem-rule-name>
{
    LocalPortGroupRef      <db2-subsystem-portgroup-name>
    JobName                <db2-subsystem-ddf-task>
    Direction              Inbound
    TTLSGroupActionRef     <db2-group-action-name>
    TTLEnvironmentActionRef <db2-subsystem-environment-name>
}
```

```

}

PortGroup <db2-subsystem-portgroup-name>
{
    PortRange
    {
        Port <db2-subsystem-secport-1>
    }
    PortRange
    {
        Port <db2-subsystem-secport-2>
    }
    ...
    PortRange
    {
        Port <db2-subsystem-secport-n>
    }
}

```

- ▶ <db2-subsystem-rule-name> is a unique identifier for the Db2 subsystem-specific rules in the AT-TLS configuration. It should not contain blanks or special characters for easier reference.
- ▶ <db2-subsystem-portgroup-name> is a unique identifier for the Db2 subsystem-specific port group in the AT-TLS configuration. It should not contain blanks or special characters for easier reference.
- ▶ <db2-subsystem-ddf-task> is the name of the Db2 subsystem's DDF address space <ssid>DIST.
- ▶ <db2-group-action-name> is the unique identifier that references the shared definition in the AT-TLS configuration to enable encryption.
- ▶ <db2-subsystem-environment-name> is the unique identifier that references the Db2 subsystem-specific environment action in the AT-TLS configuration.
- ▶ <db2-subsystem-secport-1> to <db2-subsystem-secport-n> are the secure port numbers for Db2 that should be encrypted. Multiple PortRange options can be specified, each one with a single Port number. Although PortRange also supports the definition of a consecutive range of port numbers on each Port statement, readability is easier when each port is listed individually.

## Adding special encryption settings to an existing TTLSRule

When an encryption setup for Db2 exists but does not suit the encryption requirements for IBM Integrated Synchronization, an extra **TTLSRule** specifically for IBM Integrated Synchronization purposes should be defined. Encryption details might interfere, for example, if client authorization is always done by using client certificates.

If a dedicated secure port is used for IBM Integrated Synchronization through a location alias, then these two **TTLSRule** statements are independent because of the different port numbers and do not interfere. However, if the same port must be used again for IBM Integrated Synchronization but the other encryption settings are in conflict, then a second **TTLSRule** can still be used with more specific connection properties, for example, the IP range of the instances of Accelerator on the private network. In addition, both **TTLSRule** statements get a priority assigned so that AT-TLS always picks the Accelerator-specific rule if such a specific connection comes in, and falls back to the existing rule for all other connections.

To achieve this specific setup, two **TTLRule** statements with the **Priority** options must be specified. The lowest **Priority** is 0, and **TTLRule** statements with higher **Priority** values are checked first for whether the specified parameters match. The following example describes how to use two rules for the same port number. For simplification, they show only the additional pieces that are needed:

```
TTLRule <db2-subsystem-rule-name>
{
    LocalPortRange          <db2-subsystem-secport>
    JobName                  <db2-subsystem-ddf-task>
    Direction                Inbound
    Priority                  <general-priority>
    TTLGroupActionRef        <db2-group-action-name>
    TTLEnvironmentActionRef  <db2-subsystem-environment-name>
}
```

```
TTLRule <db2-subsystem-rule-name-for-accelerator>
{
    LocalPortRange          <db2-subsystem-secport>
    JobName                  <db2-subsystem-ddf-task>
    RemoteAddr              <accelerator-ip-addr>
    Direction                Inbound
    Priority                  <accelerator-priority>
    TTLGroupActionRef        <db2-group-action-name>
    TTLEnvironmentActionRef  <db2-subsystem-env-for-accelerator>
}
```

- ▶ <general-priority> and <accelerator-priority> are numeric values where <accelerator-priority> must be greater than <general-priority>, for example, <general-priority> = 1 (which is the default) and <accelerator-priority> = 10.
- ▶ <db2-subsystem-rule-name-for-accelerator> and <db2-subsystem-env-for-accelerator> are new unique identifiers for the Accelerator-specific rules in the AT-TLS configuration. They should not contain blanks or special characters for easier reference.
- ▶ <accelerator-ip-addr> is the IP address of Accelerator for which the specific rules should be defined. It can either be a unique IP address in the format n.n.n.n or it can specify an entire subnet in the format n.n.n.n/m, where m defines the number of bits that define the subnet, for example, 192.168.4.0/24.

## Activating AT-TLS configuration changes

The AT-TLS configuration changes are not effective immediately. PAGENT must refresh the settings by rereading the configuration files by running the **MODIFY** command for PAGENT on the z/OS console or in SDSF:

```
/f PAGENT,REFRESH
```

## Network setup for data sharing groups

You can use DDVIPA to distribute the incoming connections to available Db2 members. If the controlled availability of the dedicated secure port for IBM Integrated Synchronization ensures that only a single member receives connections at a time, no special settings on the **VIPADISTRIBUTE** statements are needed.

However, for a full HA setup with multiple active Db2 members listening on the dedicated secure port for IBM Integrated Synchronization, you must add the **TIMEDAFFINITY** option to the **VIPADISTRIBUTE** statement. This option requires a parameter that sets the timeout for the affinity in seconds. The timeout should be set to 60, which is the timeout for log reader tasks in Db2 too.

Here an example of **VIPADISTRIBUTE** statements in the VIPADYNAMIC section:

```
VIPADYNAMIC
VIPADISTRIBUTE 1 <ddvipa-netmask> <ddvipa-ip-address>
                DEFINE TIMEDAFFINITY 60 <ddvipa-ip-address>
                PORT    <db2-integrated-synchronization-secport>
                DESTIP  <db2-member-ip-address-1>
                        <db2-member-ip-address-2>
                        ...
                        <db2-member-ip-address-n>
VIPABACKUP      1 <ddvipa-ip-address>
VIPADISTRIBUTE 1 <ddvipa-netmask> <ddvipa-ip-address>
                DEFINE TIMEDAFFINITY 60 <ddvipa-ip-address>
                PORT    <db2-integrated-synchronization-secport>
                DESTIP  <db2-member-ip-address-1>
                        <db2-member-ip-address-2>
                        ...
                        <db2-member-ip-address-n>

VIPABACKUP      2 <ddvipa-ip-address>
VIPADISTRIBUTE 2 <ddvipa-netmask> <ddvipa-ip-address>
                DEFINE TIMEDAFFINITY 60 <ddvipa-ip-address>
                PORT    <db2-integrated-synchronization-secport>
                DESTIP  <db2-member-ip-address-1>
                        <db2-member-ip-address-2>
                        ...
                        <db2-member-ip-address-n>
```

- ▶ **<ddvipa-netmask>** is the netmask that applies to the interface on which the DDVIPA should listen to incoming connections.
- ▶ **<ddvipa-ip-address>** is the IP address where the DVIPA listens for incoming connections and then distributes the requests to active Db2 members.
- ▶ **<db2-integrated-synchronization-secport>** is the dedicated secure port number that is defined in the location alias.
- ▶ **<db2-member-ip-address-1>**, **<db2-member-ip-address-2>**, and to **<db2-member-ip-address-n>** are the IP addresses of the Db2 members that can take requests for IBM Integrated Synchronization and where the location alias is defined.

## Db2 setup

This section describes the Db2 setup for the environment in this paper.

### Installing the required maintenance and verifying the function level

IBM Integrated Synchronization is procured through an APAR for Db2 V12 for z/OS, so the PTF for this APAR must be installed. It is a best practice to install all existing Db2 maintenance for IBM Integrated Synchronization.

All Db2 for z/OS PTFs that are related to IBM Integrated Synchronization require a stop and restart of the Db2 subsystem to become active.

A minimum function level of V12R1M500 is required to use IBM Integrated Synchronization.

### Granting Db2 privileges to read log records and stored procedure execution

“Security setup in RACF” on page 18 suggests creating a dedicated user ID that is used by IBM Integrated Synchronization to authenticate with Db2. The RACF permits for this user ID are described in that section, but extra privileges in Db2 must be granted so that it can read log records and start stored procedures to react to table-specific replication errors.

The following **GRANT** statement in Db2 allows the incremental update user ID to access log records:

```
GRANT MONITOR2 TO <int-synch-user>;  
COMMIT;
```

<int-synch-user> is the user ID that is created for incremental updates through IBM Integrated Synchronization.

In addition, this user ID requires the runtime privilege on the stored procedure ACCEL\_SET\_TABLES\_ACCELERATION, which is started when an error is experienced during incremental updates that takes a table out of replication. By calling that stored procedure, the table is set as disabled for query acceleration to avoid stale data being returned when current data is expected. This privilege is granted by using the following Db2 statement:

```
GRANT EXECUTE ON PROCEDURE SYSPROC.ACCEL_SET_TABLES_ACCELERATION  
TO <int-synch-user>;  
COMMIT;
```

<int-synch-user> is the user ID that is created for incremental updates through IBM Integrated Synchronization.

The authorization for the stored procedures is verified when incremental updates through IBM Integrated Synchronization is enabled on Accelerator through the configuration console. If the privileges are missing, an error is displayed and incremental updates cannot be enabled. After you fix the missing privileges, then incremental updates can be enabled.



## Defining a secure port for encrypted connections to Db2

Db2 for z/OS allows both unencrypted and encrypted connections that are distinguished by using different port numbers on which Db2 is listening. These port numbers are defined in the DDF settings and can be displayed by using the **-DISPLAY DDF** command:

```
DSNL080I  -DB2A DSNLTDDF DISPLAY DDF REPORT FOLLOWS:
DSNL081I  STATUS=STARTD
DSNL082I  LOCATION          LU-name          GENERICLU
DSNL083I  DB2ALOC          NATIVE.APPLDB2A    -NONE
DSNL084I  TCPPORT=8010     SECPORT=18010     RESPOR=8011  IPNAME=-NONE
...
```

This example shows that Db2 already is configured with a secure port. This secure port is configured in the Bootstrap Data Set (BSDS) or by using the system parameter **DRDA SECURE PORT**, which is specified on the DSNTIP5 panel. For more information about setting the SECPORT in the BSDS by using the DSNJU003 change log inventory utility, see [IBM Knowledge Center](#).

This general Db2 secure port can be used for IBM Integrated Synchronization, but it is a best practice to use a dedicated secure port.

### Defining a location alias for dedicated secure ports

Db2 supports location aliases in DDF that can have individual connection properties, for example, a dedicated SECPORT for remote communication. Such a location alias is the solution to provide the dedicated secure port for IBM Integrated Synchronization. The location alias definition is independent from an existing SECPORT definition, except that it cannot have the same secure port number. In fact, the general Db2 SECPORT does not need to be defined when IBM Integrated Synchronization is set up. In that case, it is sufficient to create the location alias with its own SECPORT because general access to Db2 would not be using secure connections.

The location alias can be set up in two ways:

- ▶ Using an online **MODIFY** command.
- ▶ Adding the location alias to the definition in the BSDS.

The MODIFY command activates the location alias immediately. The BSDS method has the disadvantage that Db2 must be stopped and restarted to activate the change.

For more information about both methods, see the following resources:

- ▶ The MODIFY DDF command at [IBM Knowledge Center](#)
- ▶ The DSNJU003 (change log inventory) utility at [IBM Knowledge Center](#)

The following Db2 commands can be used to create a location alias with a dedicated secure port and start it for immediate activation:

```
-MODIFY DDF ALIAS(<alias-name>) ADD
-MODIFY DDF ALIAS(<alias-name>) SECPORT(<db2-subsystem-secport>)
-MODIFY DDF ALIAS(<alias-name>) START
```

- ▶ <alias-name> is the name of the location alias that is created for IBM Integrated Synchronization. It can be up to 16 characters long.
- ▶ <db2-subsystem-secport> is the port number that is used for the dedicated secure port. It must be unique on the LPAR or in the SYSPLEX so that it can be reached correctly by Accelerator.

Location aliases are specific to a Db2 subsystem. In data sharing groups, they must be added to each member that must provide a secure port for IBM Integrated Synchronization.

### **Managing secure port availability with location aliases**

An advantage of location aliases is that they can be started and stopped individually per Db2 member. They can be used to control resources and availability of the secure port for IBM Integrated Synchronization on each Db2 member in a data sharing group, for example, for planned maintenance. Other distributed workloads that access the general Db2 secure port would not be affected by stopping a location alias.

To stop receiving incoming connections on the secure port that is defined by the location alias, run the following Db2 command:

```
-MODIFY DDF ALIAS(<alias-name>) STOP
```

To enable new connections to the secure port, run the following command:

```
-MODIFY DDF ALIAS(<alias-name>) START
```

<alias-name> is the name for the location alias for IBM Integrated Synchronization.

## **Accelerator setup for incremental updates**

The final steps to activate incremental updates by using IBM Integrated Synchronization are needed on Accelerator itself.

### **Transferring the public key for the CA certificate to Accelerator**

The public key for the CA certificate was exported to a data set in “Security setup in RACF” on page 18. This public key now must be transferred to Accelerator so that the Db2 server certificate can be verified that it was signed by this CA certificate.

To complete this transfer, in IBM Db2 Analytics Accelerator Studio, complete the following steps:

1. Open the Accelerator view of Accelerator to which the CA certificate must be transferred.
2. Click **Encryption details**.
3. Click **Transfer new certificates**.
4. Depending on whether the CA certificate is available in HFS or on the workstation, click **Transfer file from System Z...** or **Transfer file from client....**
5. Go to the certificate file that is stored in HFS or on the workstation and select it.
6. Click **Transfer**.

## Enabling incremental updates on Accelerator

Incremental updates are enabled and disabled in IBM Db2 Analytics Accelerator Console. Every instance of Accelerator that is paired to Db2 can use only one method for incremental updates, which is selected during the enablement process. If incremental updates are already enabled by using InfoSphere Data Replication, then this enablement must be disabled and removed *before* selecting and enabling incremental updates by using IBM Integrated Synchronization. When disabling incremental updates, all replicated tables remain on Accelerator and can still be used for queries. However, if incremental updates are enabled again, any table that is added to replication must be reloaded.

To enable incremental updates, complete the following steps:

1. Start and log on to the IBM Db2 Analytics Accelerator Console. For more information, see [Logging on to the IBM Db2 Analytics Accelerator Console in IBM Knowledge Center](#).
2. Type the number in front of the option Manage Incremental Updates and press Enter to display the following submenu:

```
main -> manageIU
```

```
-----  
You have the following options:
```

- (0) - Go back one level
- (1) - Enable incremental updates
- (2) - Disable incremental updates
- (3) - Restart incremental update processes
- (4) - Include or exclude tables not enabled for incremental updates in WAITFORDATA queries

3. Type 1 and press Enter to enable incremental updates.
4. If Accelerator is paired to more than one Db2 subsystem, select the target Db2 from the list. Otherwise, the only applicable Db2 subsystem is selected automatically.
5. Type 1 and press Enter to select the incremental update technology as Integrated Synchronization.
6. Enter the connection details for Db2. The IP address of either the Db2 subsystem LPAR, or the DDVIPA group IP address and the dedicated secure port for IBM Integrated Synchronization, are needed.
7. Select the authentication method. If PassTicket authentication is not used, type N and press Enter, followed by the user ID and password that were prepared for IBM Integrated Synchronization during the RACF setup. If PassTicket authentication is used, type Y and press Enter, followed by the user ID, the Db2 application name, and the secret token (secure sign-on key) that was used in “Optional: Using PassTickets with Db2” on page 18.
8. Select the CA certificate that was transferred to Accelerator that will be used to verify the Db2 server certificate. All certificates that are found in the transfer directory will be listed. Type the number of the CA certificate and press Enter to select it.
9. Type Y and press Enter to configure the Db2 subsystem for replication.
10. Accelerator connects to Db2, verifies the credentials and privileges, and sets up incremental updates.
11. Press Return after the configuration completes with a Done message.

## Starting replication

After incremental updates are enabled, replication is in the Stopped state by default. You can start replication again by going to the **Accelerator** view of IBM Db2 Analytics Accelerator Studio and clicking **Start**.

## Enabling tables for replication

Before a table can be enabled for replication, the `DATA CAPTURE CHANGES` attribute of the table must be enabled. In IBM Db2 for z/OS Analytics Accelerator before Version 7.5.2, this task is done by the Db2 administrator, who issues the following Db2 statement:

```
ALTER TABLE <table-name> DATA CAPTURE CHANGES;
```

<table-name> specifies the table in Db2 that will be enabled for replication.

Starting with IBM Db2 for z/OS Analytics Accelerator Version 7.5.2, this task is done automatically when the table is enabled for replication.

## Installation experiences from other customers

IBM Integrated Synchronization is designed and developed with early customer involvement to ensure that installation, operation, and performance meet the expected and wanted design goals. To validate our goals for this technology, an Early Support Program was initiated in the second half of 2019. This program enabled the responsible customer advocates together with the development team to work with the selected customers on specific areas and gather feedback about our objectives. Within this program, we wanted to understand the product usage within the customer environment and explore the following objectives:

- ▶ Simplified installation of IBM Integrated Synchronization
- ▶ Simplified administration of IBM Integrated Synchronization
- ▶ Key metrics for IBM Integrated Synchronization, including:
  - Latency
  - zIIP utilization
  - CPU costs on IBM Z
  - Overall performance data, such as throughput and synchronized rows per second.
- ▶ Enterprise-grade HTAP capabilities, focusing on:
  - Latency
  - Throughput
  - Queries per second

Especially in the area of installation and administration, we worked closely with a significant Italian customer and received valuable feedback from them about the product.

Areas that were improved due to customer feedback are:

- ▶ Messages and messaging:
  - Improved message texts.
  - Misleading messages were removed.
  - Error state messages were enhanced.
  - Message handling in data sharing environments was improved.
- ▶ Restart capabilities improvements:
  - Automatic restart due to member outage.
  - Restart time reduced to less than 1 minute.
- ▶ Certificate handling enhanced:
  - Data Studio certificate management dialog was implemented and improved.
  - Certificate validation was improved.
  - Automatic conversion of certificate format was implemented.
- ▶ HA capabilities for the log reader task were requested and implemented.

- ▶ Requirements were opened that will be implemented with upcoming product updates:
  - Disable query acceleration for replicated tables in error (available with Version 7.5.2).
  - Support of schema changes.
- ▶ Alter Table Add Column.

## Enabling tables for replication

When enabling replication for tables that are loaded on Accelerator by using IBM Integrated Synchronization, **DATA CAPTURE CHANGES** must be enabled for the table. Starting with IBM Db2 Analytics Accelerator V7.5.2, this task is done automatically. For versions before Version 7.5.2, **DATA CAPTURE CHANGES** must be enabled by the Db2 administrator or replication cannot be enabled.

Running **ALTER TABLE ... DATA CAPTURE CHANGES** might interfere with concurrent workload. Therefore, it is a best practice to have **DATA CAPTURE CHANGES** enabled before enabling replication if the table is heavily used in system workloads. By doing this task separately, the administrators may do the enablement at a suitable time that does not affect online applications.

## Replicated tables in error

If the incremental update processing of IBM Integrated Synchronization detects an error, the affected table is placed into the ERROR status, and further incremental updates for the table are suspended. A possible error situation for a table might be that log records could not be decompressed because of other concurrent workload preventing access to the decompression dictionaries, or table operations that are not supported by IBM Integrated Synchronization.

Starting with IBM Db2 Analytics Accelerator V7.5.2, a table that is removed from incremental updates also is made unavailable for query acceleration to prevent returning results from an outdated copy on Accelerator.

To re-enable replication and acceleration, the table must be reenabled for replication, which requires a reload of the table to bring it to the current level.

## IBM Integrated Synchronization externals in Db2 for z/OS

When using IBM Integrated Synchronization, every incremental update enablement between Accelerator and Db2 for z/OS starts an asynchronous log reader task to collect log records for the tables that must be synchronized. These log reader tasks are run in zIIP eligible system request blocks (SRBs) in the DBM1 address space. The log reader tasks remain active while Accelerator is connected to Db2. If for any reason a connection is dropped unexpectedly, Db2 cleans up any log reader task that has not been referenced for 60 seconds.

When a new log reader task is started, message DSNI090I is issued:

```
08.46.07 STC18562 DSNI090I -DC11 DSNILGRT -STARTING ASYNCHRONOUS LOG 753
          753 READER TASK FOR
          753 SESSION ID 43299E29761A0805
          753 STARTING AT 00000000026A1D883F4A
          753 WITH 1 QUALIFIERS
```

When a log reader task is terminated, either on request from Accelerator or through a timeout, message DSNIO92I is issued:

```
08.46.48 STC18562 DSNIO92I -DC11 DSNILGRT -NORMAL TERMINATION OF 772
              772 ASYNCHRONOUS LOG READER TASK FOR
              772 SESSION ID 43299E29761A0805
              772 ENDING AT 00000000026A1D89405E
              772 AFTER READING 92 LOG RECORDS
```

If any error is encountered during log reading, the log reader task is terminated after issuing message DSNIO91I. The z/OS system log should include more information about the failure, for example, an invalid relative byte address (RBA) that is specified by Accelerator that cannot be found in any active or archive log data set.

```
08:35:43.48 STC07989 DSNIO90I -DC11 DSNILGRT -STARTING ASYNCHRONOUS LOG 306
              306 READER TASK FOR
              306 SESSION ID 523DACEB8EBA0801
              306 STARTING AT 00000000026A1D9B6822
              306 WITH 1 QUALIFIERS
08:35:43.56 STC07989 DSNJ104I -DC11 DSNJR206 RECEIVED ERROR STATUS 307
              307 00000004 FROM DSNPCLOC FOR
              307 DSNNAME=DSNC11.ARCHLOG1.D19163.T1631487.A0019952
08:35:43.56 STC07989 DSNJ113E -DC11 DSNJR003 RBA '00000000026A1D9B6000' 308
              308 NOT IN ANY ACTIVE OR ARCHIVE LOG DATA SET.
CONNECTION-ID=DC11,
              308 CORRELATION-ID=014.LGRTSK01, MEMBER-ID=0
08:35:43.56 STC07989 DSNIO91I -DC11 DSNILGRT -ERROR IN ASYNCHRONOUS LOG 309
              309 READER TASK FOR
              309 SESSION ID 523DACEB8EBA0801
              309 RETURN CODE 8
              309 REASON 00C9000A
08:35:43.56 STC07989 DSNIO92I -DC11 DSNILGRT -ERROR CAUSED TERMINATION OF 310
              310 ASYNCHRONOUS LOG READER TASK FOR
              310 SESSION ID 523DACEB8EBA0801
              310 ENDING AT 00000000026A1D9B6822
              310 AFTER READING 0 LOG RECORDS
```

The status of current log reader tasks can be shown by running either of the following **DISPLAY STATS** commands:

- ▶ **-DISPLAY STATS(LOGREADERTASKS)**
- ▶ **-DISPLAY STATS(LRT)** (short)

The command displays all log reader tasks that are collecting log records in the current Db2 subsystem:

```
DSNT788I -DB2A
SESSIONID      STATUS  CURR. POSITION NUM  RECS AGE
-----
164FA983947B0801 SUSP EOS 0000000000018898F744 4 167s
***** DISPLAY OF STATS TERMINATED *****
DSN9022I -DB2A DSNTDSTS 'DISPLAY STATS' NORMAL COMPLETION
```

The following STATUS values may be shown for log reader tasks:

- ▶ **RUNNING:** Currently processing log records
- ▶ **READING:** Reading log records from log

- SUSP READ: Task suspended due to full buffer, and waiting for Accelerator to fetch more records
- SUSP EOS: Task suspended at end of log, and task will be woken up in short intervals to check for more log records
- CANCEL: Task in termination processing

## Coexistence with InfoSphere Change Data Capture

IBM Db2 Analytics Accelerator supports both IBM Integrated Synchronization and InfoSphere CDC, so you can perform incremental updates with either of them, that is, you can replicate from one Db2 subsystem to one Accelerator through InfoSphere CDC and to another one through IBM Integrated Synchronization. However, you cannot replicate to one attached Accelerator through both technologies in parallel (only one technology per attached pairing).

Figure 12 shows an environment where two different Db2 for z/OS subsystems replicate data to a single Accelerator. One LPAR is using IBM Integrated Synchronization, and the second one is running with InfoSphere CDC.

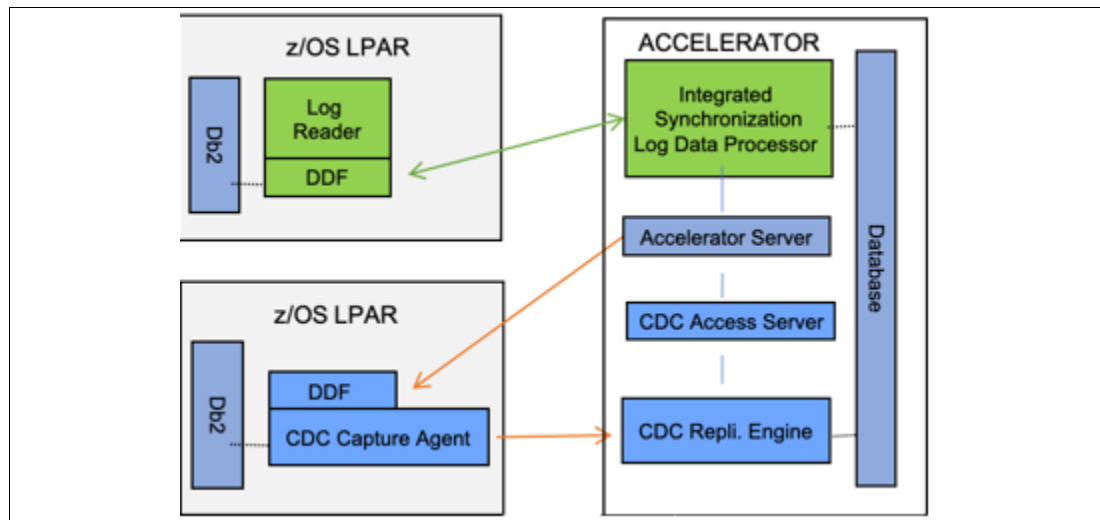


Figure 12 Connectivity options for InfoSphere CDC and IBM Integrated Synchronization for the same Accelerator

Figure 13 is a typical scenario with two Accelerators during migration, where Accelerator 1 is a Version 7 Accelerator replicating with IBM Integrated Synchronization, and Accelerator 2 is a Version 5 Accelerator still running for a certain period during the migration phase with InfoSphere CDC.

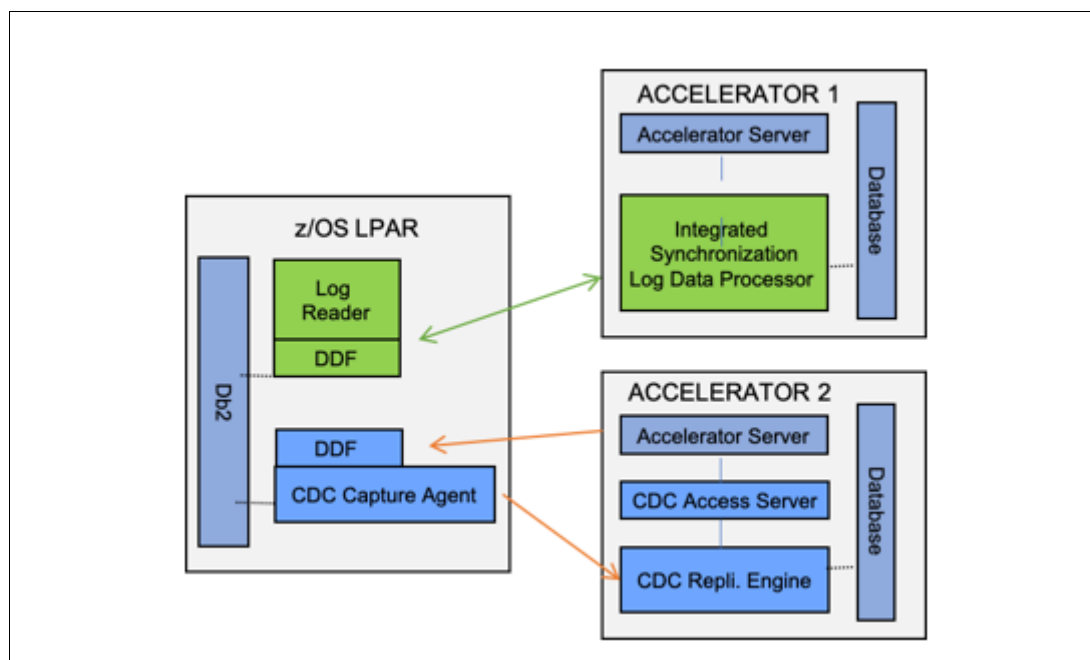


Figure 13 Connectivity options for a mixed environment

Mixed environments (see Figure 12 on page 37) are specifically of interest in coexistence scenarios where, for example, the production Db2 for z/OS system is still running with InfoSphere CDC and the test Db2 for z/OS could be configured with IBM Integrated Synchronization.

Here are examples of such environments:

- ▶ The production LPAR is running on Db2 for z/OS V11 and replicating through InfoSphere CDC, and another Db2 subsystem is migrated and running on Db2 for z/OS V12 with all prerequisites in place and could use IBM Integrated Synchronization to replicate data to the Accelerator.
- ▶ A Db2 subsystem is defined as a test system and is running IBM Integrated Synchronization for test purposes, and the Db2 production LPAR is still using InfoSphere CDC for replication. After a certain period, the production LPAR also is migrated to IBM Integrated Synchronization.

## Migrating from InfoSphere Change Data Capture

This section describes the major steps to migrate from InfoSphere CDC to IBM Integrated Synchronization. In this context, two major migration scenarios are relevant:

- ▶ Migration from Accelerator V5 with InfoSphere CDC to Accelerator V7 with IBM Integrated Synchronization
- ▶ Migration from InfoSphere CDC to IBM Integrated Synchronization

HA considerations are irrelevant because they are setup-specific.

Table 1 on page 39 show some differences before and after the migration



*Table 1 Migrating from InfoSphere CDC to IBM Integrated Synchronization*

<b>Function</b>	<b>InfoSphere CDC</b>	<b>IBM Integrated Synchronization</b>
Db2 for z/OS Transparent Archive and Replication to the Accelerator.	Does not work with Version 7 and InfoSphere CDC.	Works with Version 7 and IBM Integrated Synchronization.
Replication of non-archived HPSS partitions.	Works with InfoSphere CDC in Version 5.	Does not work with Version 7 and IBM Integrated Synchronization.
<b>ALTER TABLE</b> schema changes.	Works with Version 5.	Not supported for Version 7 and IBM Integrated Synchronization.

### **Migrating from Accelerator V5 with InfoSphere CDC to Accelerator V7 with IBM Integrated Synchronization**

In this case, a Db2 subsystem or a Db2 Data Sharing Group is running with Db12 function level 500 and is paired with Db2 Analytics Accelerator V5 replicating data through InfoSphere CDC. A second Accelerator is paired running Db2 Analytics Accelerator V7. The system is configured in coexistence mode so that the Db2 subsystem can access both Accelerators running in different Accelerator versions, meaning two Accelerators (Version 5 and Version 7) are paired to the same subsystem.

In this scenario, start with IBM Integrated Synchronization to avoid extra costs and overhead. In addition, there is no downtime for Accelerator V7, which allows continuous operation over the migration period. Using IBM Integrated Synchronization on Accelerator Version 7 is the preferred method for migration instead of migrating from CDC to CDC and then redoing everything with IBM Integrated Synchronization.

### **Migrating from InfoSphere CDC to IBM Integrated Synchronization**

In this case, a Db2 subsystem or a Db2 Data Sharing Group is running with Db12 Function level 500 and paired with Db2 Analytics Accelerator V7 replicating data through InfoSphere CDC. HA considerations are irrelevant because they are setup-specific details that do not affect migration.

Figure 14 shows the setup, including the private network and port definitions.

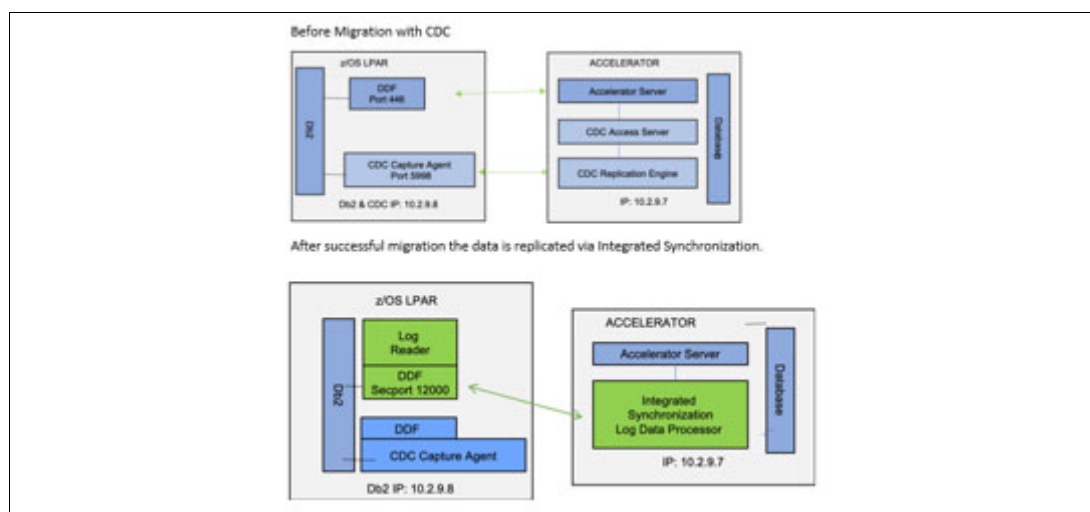


Figure 14 Sample setups for migration from InfoSphere CDC to IBM Integrated Synchronization

During the switch phase from InfoSphere CDC to IBM Integrated Synchronization, no data is replicated to the Accelerator. The time window within the migration phase to switch from InfoSphere CDC to IBM Integrated Synchronization must be at least as long as it takes to reload all affected tables on the Accelerator.

Complete the following steps:

1. Disable InfoSphere CDC:
  - a. Stop replication for the subsystem and disable all affected tables from replication through Data Studio.
  - b. Disable replication for the subsystem through the Configuration console.
2. Enable IBM Integrated Synchronization:
  - a. Enter the required parameters (Db2 IP address, secure port number, the user ID for Db2 access, and the signer certificate) for the subsystem to enable replication (IBM Integrated Synchronization) through the Configuration console.
  - b. Start replication for the subsystem through the Configuration console.
  - c. Ensure that data capture changes are enabled for all affected tables (this step is required only when running with an Accelerator software level earlier than Version 7.5.2).
  - d. Reload all affected tables.

After a certain period of replicating data with IBM Integrated Synchronization, remove the InfoSphere CDC agent on the affected Db2 for z/OS LPARs.

## REORG DISCARD and LOAD DD DUMMY support

Starting with Version 7.5.1, IBM Integrated Synchronization supports **REORG DISCARD** operations where all rows in a partition are discarded. Only Partition by Range table spaces are supported. Partition by Growth or Non-Partitioned table spaces are not supported.

The partitions that are affected must be empty or no data is discarded. No data is allowed to move between partitions (there is no support for altered limit keys). If these conditions are met, then no reload of the affected partitions is necessary, and IBM Integrated Synchronization deletes the rows from those partitions on Accelerator.

Similarly, Version 7.5.1 added support for **LOAD DUMMY** to empty quickly a table space. Again, table spaces must be partitioned by range and the **LOAD** can be for the table space level or partition level.

For example, assume that a table is partitioned as follows, and then enabled for replication and loaded to Accelerator:

```
PARTITION BY (C1)
(PARTITION 1 ENDING AT (100),
 PARTITION 2 ENDING AT (200),
 PARTITION 3 ENDING AT (300),
 PARTITION 4 ENDING AT (400),
 PARTITION 5 ENDING AT (500))
```

After the table is reorganized with **LOG NO DISCARD FROM TABLE XYZ WHEN (C1 < 200)**, data in the first two partitions is deleted both in Db2 and Accelerator.

If **REORG** is run with **WHEN (C1 < 180)**, which is a partial discard, data is not deleted from Accelerator. Replication stops for the table, and a table reload is required to get out of the replication **ERROR** state.

## ALTER TABLE changes through log records

At the time of writing, **ALTER TABLE** support is not in Version 7.5.2. This function is still an open item.

Schema Change features will be reflected in phases. In stage 1, they will be equivalent to InfoSphere CDC by tolerating compatible DDL changes. In stage 2, transactionally consistent DDL replication for mirrored tables will be provided. At the time of writing, the design of this feature is still under discussion.

## Monitoring performance in Db2 for z/OS

Because of InfoSphere CDC based replication, there are counters for monitoring in ICFID 2.

One of the easiest ways to monitor replication in Db2 for z/OS is to use the **-DIS ACCEL** command. When you run the command, you should focus on a few lines:

► CURRENT REPLICATION LATENCY FOR THIS Db2 SYSTEM = XXXX MS

This line shows the current replication latency for this Db2 for z/OS system. It shows how far the target Accelerator is behind the source. Latency is defined as the time difference between the timestamp of the last log record that was applied to the target compared to the current time.

► TOTAL CPU FOR REPLICATION FOR THIS Db2 SYSTEM = XXXX MS

This line shows the total CPU cost for data replication for this Db2 for z/OS subsystem.

In the Data Studio main control panel monitoring section, you can see the CPU cost of three different workloads on Accelerator: query execution, data maintenance, and replication. The CPU cost of replication for this Db2 subsystem also can be monitored from that panel.

Starting with IBM Integrated Synchronization, IFCID 2 has new counters. Most of the CPU for IBM Integrated Synchronization in Db2 for z/OS is spent while reading the log records for the replicated tables within the asynchronous log reader tasks. APAR PH18334 enhances statistics record 2 to report all the CP, zIIP, and zIIP on CP times that accumulated in all log reader tasks since Db2 started. APAR PH26681 further enhances IFCID 2 to report CP, zIIP, and zIIP on CP times for each active Accelerator instance.

Three fields were added to the data section QIST:

```

/* IBM Integrated Synchronization asynchronous log    @334*/
/* reader task statistics accumulated over all tasks @334*/
/* since Db2 start.                                @334*/
3 QISTLRCP CHAR(8), /* CPU time                      @334*/
3 QISTLRZI CHAR(8), /* zIIP time                      @334*/
3 QISTLRZE CHAR(8), /* zIIP eligible time                 @334*/
3 *          CHAR(24), /* For future use                      @334*/
2 QISTEND CHAR(0); /* END OF DM STATISTICS BLOCK */

```

PH26681 provides more fields in the data section Q8ST for each active Accelerator:

```

2 Q8STLRCP CHAR(8),      ! CPU time used by Integrated
                        ! Synchronization asynchronous log
                        ! reader task on behalf of this
                        ! accelerator
2 Q8STLRZI CHAR(8),      ! zIIP time used by Integrated
                        ! Synchronization asynchronous log
                        ! reader task on behalf of this
                        ! accelerator
2 Q8STLRZE CHAR(8),      ! zIIP eligible time for Integrated
                        ! Synchronization asynchronous log
                        ! reader task on behalf of this
                        ! accelerator
2 *          CHAR(80),    ! mapping for extension
2 *          CHAR(0);

```

## Authors

This paper was produced by a team of specialists from around the world.

**Christian Michel** is a software engineer at the IBM Germany development lab in Boeblingen with more than 20 years experience in the Db2 for z/OS environment. He was part of the Db2 for z/OS utilities development team for almost 15 years, and now focuses on Db2 for z/OS development in the scope of IBM Db2 Analytics Accelerator. Christian regularly speaks about his work at conferences, and enjoys working with customers to use their experiences to improve usability and understand their needs.

**Cüneyt Göksu** works as Executive IT Specialist at the IBM Germany development lab in Boeblingen. He has more than 25 years of ongoing experience with Db2 for z/OS. He was the Technical Leader of Data and AI on Z in IBM MEA before joining IBM Germany. He primarily works on Db2 Analytics Accelerator proofs of concepts, customer deployments, migrations, resolution of critical situations, and he and enjoys working on diverse projects with customers. Before joining IBM Germany, he was a member of the IBM Db2 Gold Consultant Team. He was an IBM Champion and worked as a regular instructor for IBM Education Programs. He is a L3 Certified IT Specialist and holds a PhD in computer science.

**Günter Schöllmann** works for the Analytics on IBM Z Center of Excellence as a Software Engineer and PMP. He is an PMI certified Project Manager and IT specialist working at the Analytics on IBM Z Center of Excellence team at the IBM Research® and Development Lab in Boeblingen, Germany. As part of the development organization, his responsibilities include customer consultation and customer advocacy, proofs-of-concepts, product deployments, migration support, resolution of critical situations, and technical education and documentation of IBM Db2 Analytics Accelerator. During his more than 35 years with IBM, he has worked as a developer, team leader, customer advocate, and project manager in Db2 for z/OS® utilities and other information management products.

## Now you can become a published author too

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks® residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

Find out more about the residency program, browse the residency index, and apply online at:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Stay connected to IBM Redbooks

- Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>

- Stay current on recent Redbooks publications with RSS Feeds:

<http://www.redbooks.ibm.com/rss.html>



# Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.


## COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, 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 “Copyright and trademark information” at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

Db2®	IBM Research®	Redbooks®
IBM®	IBM Z®	Redbooks (logo)  ®
IBM Cloud®	InfoSphere®	z/OS®
IBM Cloud Pak®	RACF®	

The following terms are trademarks of other companies:

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Other company, product, or service names may be trademarks or service marks of others.







REDP-5616-00

ISBN 0738459283

Printed in U.S.A.

Get connected

