SAP Landscape Transformation Replication Server

Performance Optimization Guide

Typographic Conventions

Type Style	Description
Example	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options. Textual cross-references to other documents.
Example	Emphasized words or expressions.
`EXAMPLE`	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
`Example`	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
`Example`	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
`<Example>`	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
`EXAMPLE`	Keys on the keyboard, for example, `F2` or `ENTER`.

Document History

Version	Date	Change
1.0	2015-10-26	Initial version.
1.1	2015-06-10	Updated version to reflect new SAP Note numbers.
1.2	2016-11-14	Updated version to reflect enhancements to examples.
1.3	2017-09-29	Updated version to reflect support for Sybase ASE.
1.4	2018-04-19	Knowledge base article 2082199 replaced with 2540300.
1.5	2018-05-09	Information about reading types updated.
1.6	2018-09-27	Information added about field FILTERSET from table DMC_ACSPL_SELECT.
1.7	2019-01-06	Information about filters updated.
1.8	2020-10-07	Updated to reflect both recording mechanisms.

Table of Contents

1 General Information

This guide is valid for SAP Landscape Transformation Replication Server version 2.0 SP9 (DMIS 2011 SP9) or higher. If you are running a lower release, some functions might not be available.

The information described in this guide is different depending on the recording mechanism used. If the source system is an SAP S/4HANA on-premise system (release 2020 or higher), then the subscription-based change data capture mechanism is used to record changes. Otherwise, the legacy recoding mechanism is used.

2 The Initial Load (Legacy Recording Mechanism)

You use the initial load to load data from the source system to the target system. You can do this in the SAP LT Replication Server Cockpit (transaction `LTRC`) by choosing *Data Provisioning*, and then *Start Load.* You can also use the option *Start Replication*, which starts the initial load and then begins the replication process.

2.1 Reading Types

Reading data in a system depends on the underlying database. SAP LT Replication Server provides different options for reading data from source systems. These different options are called reading types, and are described in more detail in the sections below.

A default reading type is used for a configuration. This means that data will be read from the source system database using this reading type for all tables in the configuration. When you create a configuration, you specify an initial load mode. The reading types associated with each option are outlined in the sections below

2.1.1 Source System is Connected to SAP LT Replication Server System by a Database Connection

Initial Load Mode	Default Reading Type
Resource Optimized	1 - *Range Calculation*
Performance Optimized	5 - *Sender Queue*

2.1.2 Source System is Connected to SAP LT Replication Server System by an RFC Connection

Initial Load Mode	Default Reading Type
Resource Optimized	3 - *Ordered by Primary Key* (If the source system is running on an SAP HANA database, then reading type 1 - *Range Calculation* is always used.)
Performance Optimized	Transparent Tables: 5 - *Sender Queue* Pool / Cluster Tables: 4 - *Sender Queue (Cluster Tables)*

In addition to the default reading type you can specify a reading type for specific tables. You can do this in *Advanced Replication Settings* (transaction LTRS) under *Performance Options*.

If it is acceptable to temporarily use some additional storage in the source system for the initial load, we recommend using the initial load mode *Performance Optimized* as this is the most efficient way of loading the data.

2.1.3 Reading Type 1 - Range Calculation

For reading type 1, *Range Calculation*, the data is split into portions which are then loaded into the target system. The system uses one key field to calculate the portion. You specify a key field for which not too many records have the same value in that field. In *Advanced Replication Settings* (transaction LTRS), you can specify reading type 1 for a table. You enter the name of the field that you want to use in the field *Selection Parameter*.

Example

You have the following data in your system:

KEY_1	KEY_2	KEY_3	DATA
100	2500	1000000120	Document 1
100	3750	1000000127	Document 8
100	8000	1000000123	Document 4
100	8000	1000000125	Document 6
200	2500	1000000122	Document 3
200	2500	1000000124	Document 5
200	8000	1000000121	Document 2
200	8000	1000000126	Document 7

The following table illustrates the effect of using reading type 1, *Range Calculation* for the sample data above using the selection parameter KEY_3:

Portion	Data
1	KEY_3 <= 1000000121
2	KEY_3 > 1000000121 KEY_3 <= 1000000123
3	KEY_3 > 1000000123 KEY_3 <= 1000000125
4	KEY_3 > 1000000125

This reading type is the default reading type if the source system is running on SAP HANA database.

For all other databases, this reading type will only work properly if an index exists for the field that is defined as the selection parameter. The field used as the selection parameter must be the first field in the index.

If the first key field cannot be used for range calculation, a separate index for key field defined as the selection parameter is required. If you are loading data from all clients in a system to one target system (which is the default behavior), usually a separate index is required as the *Client* field will have too many records with the same value. If you activate the *Read from Single Client* option, you can use the key field after the *Client* field and do not need to create a separate index for the table.

This reading type can be used for ABAP-based SAP source systems, and for non ABAP-based source systems.

Technical Settings

The default portion size for reading type 1 is 8 million bytes. As the portion size is calculated by multiplying the total bytes of all fields by the number of records in a portion, actual portion size is typically smaller as not all fields are filled.

You can change the default portion size in *Advanced Replication Settings* (transaction LTRS) under *Performance Options* by using the field *Package Size*.

If the selection parameter is a field that has too many records with the same value, SAP LT Replication Server allows the portion size to be increased to 800 million bytes. If this limit is exceeded, the system returns an error stating that the selection parameter is not suitable. You have to define another field as the selection parameter, or use a different reading type.

2.1.4   Reading Type 3 - Primary Key Order

Reading type 3, *Primary Key Order* is the default reading type. It uses a function to read the records of the source table ordered by the primary key. This means that no additional index is required as the primary index is used. The function returns a set of table records (the default is 10,000 records).

Example

You have the following data in your system:

KEY_1	KEY_2	KEY_3	DATA
100	2500	1000000120	Document 1
100	3750	1000000127	Document 8
100	8000	1000000123	Document 4
100	8000	1000000125	Document 6
200	2500	1000000122	Document 3
200	2500	1000000124	Document 5
200	8000	1000000121	Document 2
200	8000	1000000126	Document 7

The following table illustrates the effect of using reading type 3, *Primary Key Order* for the sample data above:

Portion	KEY_1	KEY_2	KEY_3
1	100	2500	1000000120
	100	3750	1000000127
2	100	8000	1000000123
	100	8000	1000000125
3	200	2500	1000000122
	200	2500	1000000124
4	200	8000	1000000121
	200	8000	1000000126

This reading type does not need any additional indexes or temporary storage in the source system, but only works properly for smaller tables or for databases where data is stored in the database in the order of the primary key (for example Microsoft SQL Server).

This reading type can only be used for ABAP-based SAP source systems.

Technical Settings

The default portion size for reading type 3 is 10,000 records. You can change the default portion size in *Advanced Replication Settings* (transaction LTRS) under *Table Settings*. You can specify a value in the field *Size of Read Portions*.

In order to improve the efficiency of the initial load, several portions are grouped together in an access plan. By default, an access plan can contain up to 8 million records, which should be sufficient for most tables. This means

that each access plan contains 800 portions. There is typically no need to change the portion size, but if you do, ensure that the number is not too large as otherwise most (or even all) portions will fit into one access plan and therefore the performance of the load cannot be improved.

If required, you can change the total number of records in an access plan for reading type 3 in *Advanced Replication Settings* (transaction `LTRS`) in section *Performance Options* total by using the field *Package Size*.

Note that if reading type 3 is used to read data from a table, then you cannot specify filter conditions or range conditions (in transaction `LTRS`).

2.1.5   Reading Type 4/5 - Sender Queue

Reading type 4, *Sender Queue (Cluster Tables)* is for cluster tables. The relevant data is extracted and temporarily stored in a compressed format in a queue in the sender system. The sender system database optimizer decides on the most efficient way to extract the data. The sender queue requires additional space for the initial load (approximately 50% to 75% of the volume to be loaded).

Reading type 5, *Sender Queue* is for pool and transparent tables. The relevant data is extracted and temporarily stored in a compressed format in a queue in the sender system. The database optimizer decides on the most efficient way to extract the data. The sender queue requires additional space for the initial load (approximately 10% of the volume to be loaded).

The data in the sender queue is stored in table DMC_INDXCL. This table can grow significantly in size while an initial load is running. After the initial load is completed, the data is deleted from the table.

Example

You have the following data in your system:

KEY_1	KEY_2	KEY_3	DATA
100	2500	1000000120	Document 1
100	3750	1000000127	Document 8
100	8000	1000000123	Document 4
100	8000	1000000125	Document 6
200	2500	1000000122	Document 3
200	2500	1000000124	Document 5
200	8000	1000000121	Document 2
200	8000	1000000126	Document 7

The following table illustrates the effect of using reading type 4, *Sender Queue* for the sample data above:

Sender Queue (DMC_INDXCL) for Sample Data

CLUSTER ID	DATA
000001	0372DC93A1713B5F1AE50DAB...
000002	278D476A02E4EB60F527DD763...
000003	BDB52E893A9C0C27B5A277C3...
000004	5C8649E8822E5495D01B4A9A7...

This reading type can be used for ABAP-based SAP source systems and non ABAP-based source systems.

Technical Settings

The default portion size for reading types 4 and 5 is 8 million bytes. As the portion size is calculated by multiplying the total bytes of all fields by the number of records in a portion, the actual portion size is typically smaller as not all fields are filled.

You can change the default portion size in *Advanced Replication Settings* (transaction LTRS) under *Performance Options* by using the field *Package Size*.

By default, a single access plan calculation job copies the data from the original table to the queue in the source system. The initial load is handled by multiple data transfer jobs.

You can further improve the process that copies the data from the original table to the queue in the source system by defining number of records that are processed by one access plan. For cluster tables, this number refers to the number of records in the physical cluster table.

You can change the number of records in *Advanced Replication Settings* (transaction LTRS) under *Performance Options* by using the field *Number of Records*.

For example if there is a table with 108 million records and you specify a value of 20 million for the table, the system will generate 6 access plans which can run in parallel. The first 5 access plans will handle 20 million records each, while the last access plan will handle the remaining 8 million records. To define the value for parameter *Number of Records*, you can divide the total number of records in the table by the number of access plan that you want to run in parallel.

For cluster tables, the total number of records is related to the physical cluster table (for example RFBLG) and not the logical table that is defined in field TABNAME (for example BSEG).

The parameter *Number of Records* enables the parallel processing in the calculation phase but do not specify the number of parallel jobs that will be used. You can specify the number of calculation jobs in the *SAP LT Replication Server Cockpit* (transaction LTRC) on the tab *Administration*. By default, not more than 3 calculation jobs will process the same table. You can adjust the number of jobs that run in parallel for the same table by using *Advanced Replication Settings* (transaction LTRS). Under *Performance Options*, on the *Initial Load* tab, you can use the field *No. of Jobs* to specify the maximum number of jobs that will work on the table. In most cases, the access plans are calculated by using the primary key (there is a special behavior for transparent tables on Oracle database where the access plan is calculated based on the row ID). If the access plan is calculated based on the primary key, note that the database will most probably execute a full table scan for each calculation job. We therefore recommend using a maximum of 8 parallel jobs for the same table.

The parameter *Number of Records* must be greater than 50,000. However, we recommend using a minimum value of 10 million records.

For the initial load, system divides the table into portions. These portions are then transferred to the target system (by means of initial load jobs). Calculating the portions can take a long time for large tables.

However, if a table is particularly large (more than a billion records), you can improve the performance of the initial load by dividing the table into ranges. Dividing the table into ranges improves the performance because the job can calculate the portions for each range at the same time. You can divide the table into ranges by using *Advanced Replication Settings* (transaction LTRS). Under *Performance Options*, on the tab *Initial Load Options*, you can define ranges (choose the active view *Ranges*).

Example: Ranges defined for table BKPS

You want to split the initial load of table BKPF in mass transfer ID 001 into 4 ranges. You use field BUKRS to define the ranges:
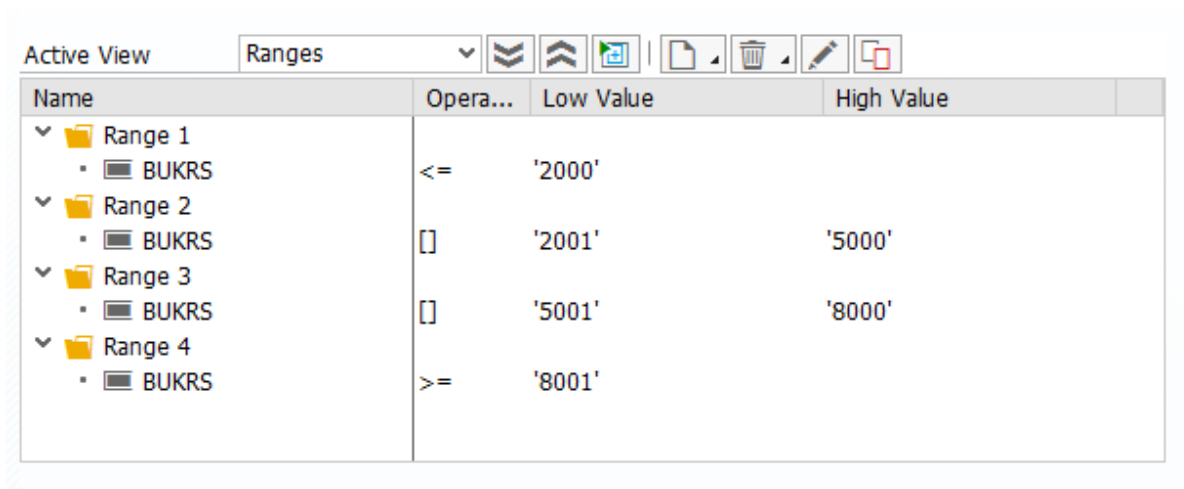Range 1: 0000 -> 2000
Range 2: 2001 -> 5000
Range 3: 5001 -> 8000
Range 4: 8001 -> 9999

You can define such ranges in the *Advanced Replication Settings* (transaction LTRS) under *Performance Options*, as depicted in the following screenshot:

Active View	Ranges				
Name		Opera...	Low Value	High Value	
⌄ 📁 Range 1					
· 🖥 BUKRS		<=	'2000'		
⌄ 📁 Range 2					
· 🖥 BUKRS		[]	'2001'	'5000'	
⌄ 📁 Range 3					
· 🖥 BUKRS		[]	'5001'	'8000'	
⌄ 📁 Range 4					
· 🖥 BUKRS		>=	'8001'		

Note

For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction LTRS.

Based on this information, SAP LT Replication Server uses the following ranges to select the data from the table:

Range 1: `SELECT FROM BKPF WHERE BUKRS <= '2000'.`

Range 2: `SELECT FROM BKPF WHERE BUKRS >= '2001' AND BUKRS <= '5000'.`

Range 3: `SELECT FROM BKPF WHERE BUKRS >= '5001' AND BUKRS <= '8000'.`

Range 4: `SELECT FROM BKPF WHERE BUKRS >= '8001'.`

If you are using ranges to improve the performance of the initial load, it is not possible to use filters for the initial load.

2.2   Improving the Performance of the Initial Load

By default, up to three jobs will work on each table. You can adjust the number of jobs that run in parallel for the same table by using *Advanced Replication Settings* (transaction `LTRS`). Under *Performance Options*, on the tab *Initial Load Options* you can use the field *Number of Jobs* to specify the maximum number of jobs that will process the table. The parallel jobs settings in *Advanced Replication Settings* are used when initial load for the table is started. If you want to adjust the settings for a running initial load, you can do this in the *SAP LT Replication Server Cockpit*. On the *Participating Objects* tab, select the relevant table and choose *Edit Object Settings*. The relevant field is called *Number of Jobs*.

The actual number of jobs that process the same table also depends on the number of running data transfer jobs. So you might need to adjust the number of transfer jobs (and initial load jobs) if required. You can set the number of transfer jobs (or initial load jobs) in *Administration Data* tab in the *SAP LT Replication Server Cockpit* (transaction `LTRC`) in edit mode*.

2.3   Filtering the Initial Load

2.3.1   Important Information

Note the following restrictions and recommendations for filters:

- The filter options are not available for tables with a name that is longer than 20 characters. The *Filter Options* tab is hidden for such tables.

- Filters for the replication process are not supported for index-like tables and cluster tables.

- If you are replicating data to SAP BW by using the ODP framework, then you cannot use filters in transaction `LTRS` (Advanced Replication Settings). However, you can specify filtering rules in transaction `LTRS` to filter data (section Rule Assignment). Note that as applying such custom rules to tables may lead to unexpected data inconsistences between the source and target system, or may negatively affect the replication process, we highly recommend engaging the services of an SLT consultant for support.

- When specifying filters for a field, you can create an index for the field to improve performance.

- Using filters on a non-key field for the initial load can negatively impact performance. You can either create an index for the field or use reading type 4 or 5 to improve performance.

- When there are filters on the non-key fields of a table, DELETE (D) and ARCHIVE (A) operations for the table are not filtered.

- If you use readying type 3 for the initial load, filter options are only available for the replication process.

2.3.2 Overview

In the SAP LT Replication system, in Advanced Replication Settings (transaction `LTRS`), you can filter the data that is transferred to the target system by using filter rules. If you use filter rules, the data is read in the source system and transferred to the SAP LT Replication system.

You can also define filters which are used when selecting the data from the source system, to avoid the transfer of unnecessary data to the SAP LT Replication Server system. You can define such filters in the *Advanced Replication Settings* (transaction `LTRS`) under *Performance Options*. Note that you cannot define filters for tables that use reading type 3, *Primary Key Order*.
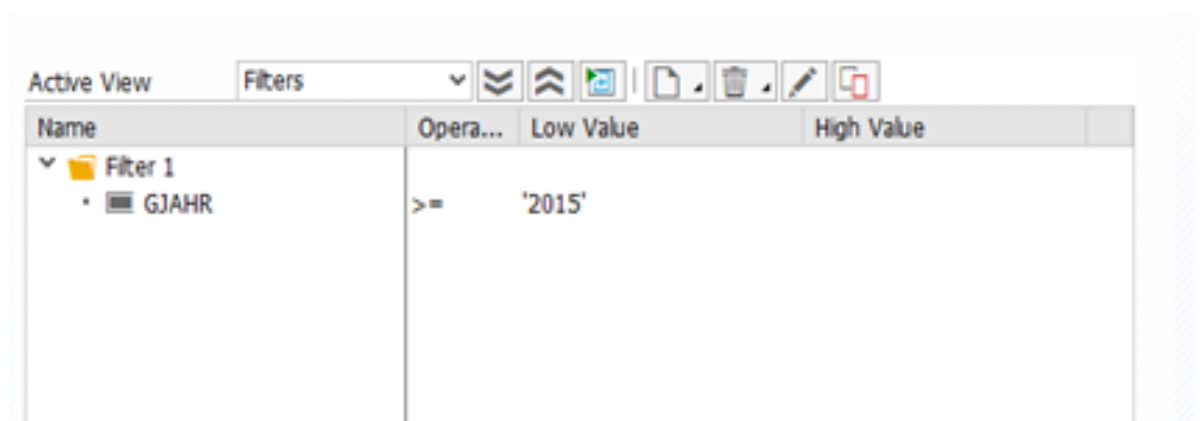
On the tab *Filter Options*, you can define filters. You can define several filters and for each filter, you can define multiple filter conditions. For each filter condition, you can define a value range for every key field such that only specific values will be replicated to the target

If you need multiple value ranges for the same key field, you need to create a separate filter for each range (as each key field can only be used once in the filter condition of one filter). All filters are used when data is selected from the source system (using an AND operator), so only those records will be transferred that fit to all defined filter conditions.

In order for any additional settings to take effect, you must save your changes in the table before you start the initial load.

Example 1: Filter for one field for a single value

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load the records where the value of the field GJAHR is greater or equal to 2015. This filter is relevant for the initial load.
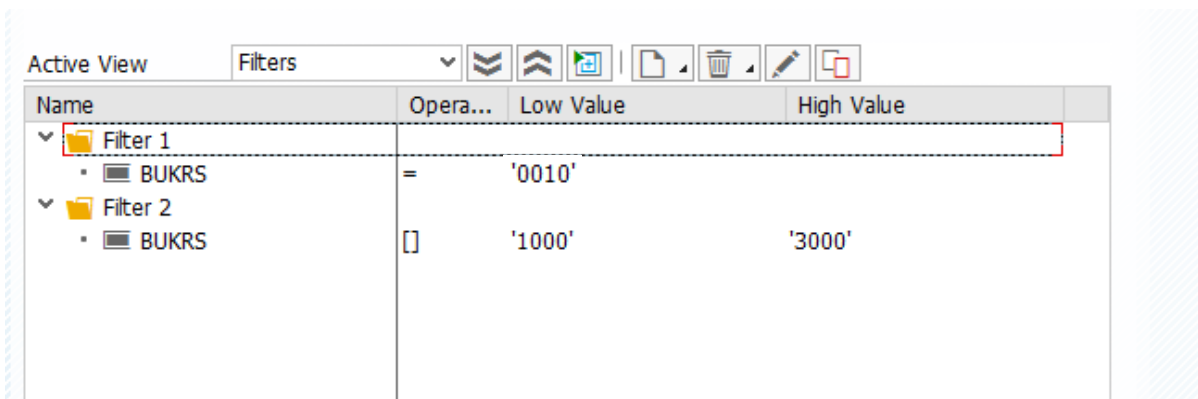
Note

> For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction `LTRS`.

Based on this information, SAP LT Replication Server uses the following WHERE clause to select the data from the table:

```
SELECT FROM BKPF WHERE GJAHR >= 2015.
```

Example 2: Filter for one field for multiple values

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load records where the value of the field BUKRS is equal 0010, or is between 1000 and 3000. This filter is only relevant for the initial load.

Note

> For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction `LTRS`.

Based on this information, SAP LT Replication Server uses the following WHERE clause to select the data from the table:

```
SELECT FROM BKPF WHERE BUKRS = '0010' OR BUKRS >= '1000' AND BUKRS <= '3000'.
```

Example 3: Filter for multiple fields

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load records where the value of the field GJAHR is greater or equal than 2015, and where the value of the field BUKRS is equal 0010 or is between 1000 and 3000. The filter is only relevant for the initial load.

Note

> For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction `LTRS`.

Based on this information, SAP LT Replication Server uses the following WHERE clause to select the data from the table:

```
SELECT FROM BKPF WHERE GJAHR >= 2015 AND BUKRS = '0010' OR BUKRS >= '1000' AND BUKRS
<= '3000'.
```

Example 4: Filter for one field using parallel jobs

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load the records where the value of the field GJAHR is greater or equal to 2015. This filter is relevant for the initial load. As the number of relevant records is more than 200 million, you want to initial load the record with 2 paralleled jobs.

| Initial Load Options | Replication Options | Filter Options |

Reading Type	5 Sender Queue ▼
Package Size	[ ] Bytes
Write Behavior	No transfer behavior was defined ▼
Number of Jobs	2
Number of Records	[ ]

3 The Replication Process (Legacy Recording Mechanism)

The replication process is used to keep data synchronized between the source system and the target system. You can trigger the replication process in the SAP LT Replication Server Cockpit (transaction `LTRC`) by choosing *Data Provisioning*, and then choosing *Start Replication*.

3.1 Processing of Logging Tables

In real-time replication mode, SAP LT Replication Server constantly checks all logging tables in the source system for new delta data. This means that all tables listed in the LT Replication Server Cockpit (transaction `LTRC`) that have the status *In Replication* are being processed by the data transfer jobs. As soon as all tables are processed, the jobs start processing all the tables again.

The default option is *Only Process Logging Tables with Records*. This means that SAP LT Replication Server identifies the logging tables that contain at least one record, and works on this list of reduced tables. If no changes occur in the source system, one replication job constantly checks for new updates, while all other data transfer jobs wait until the replication job signal that there is delta data to process.

However, if you most of the tables that have the status *In Replication* change frequently, you can change this option to the option *Process All Logging Tables*. Note that this can result in unnecessary resource consumption in the source system.

You can change this option in the LT Replication Server Cockpit. On the tab *Administration Data*, you can change the parameter *Options for Logging Tables*. If you change this option, you need to deactivate the configuration and activate it again afterwards in order for the change to take effect.

3.2 Improving the Performance of the Replication Process

SAP LT Replication Server transfers the entries from the logging table to the target system in portions. You can improve the performance of the replication by dividing the logging table into ranges. Dividing the table into ranges improves performance because the jobs can transfer the data for each range at the same time.

There are two different ways to improve the performance of the replication process for a single table. There is a generic way where changes are distributed to different ranges while they are inserted into the logging table. As this function is not available for all databases, there is an additional function that you can use to define the table ranges manually.

3.2.1   Using Generic Table Ranges

You can use table ranges to improve the performance of the replication process. The system calculates a range value in the trigger code, and stores the respective range criteria in the logging table. If this functionality is activated, the logging table will get an additional field called IUUC_REPL_GROUP, and the trigger code is enhanced in such a way that, based on the defined grouping fields, a hash value is calculated and stored in the new field. As the settings impact the logging table and the database trigger, the settings should be defined before the replication process is started for the table. If the table is being replicated, the existing logging table and trigger can be adjusted by using SAP Knowledge Base Article 2540300.

The settings can be done in the *Advanced Replication Settings* (transaction LTRS) under *Performance Options*. On the tab *Replication Options*, you need to choose *Use Key Fields to Calculate Ranges*. If this entry is not available, then your source system database does not provide the functionality to enable this option (this option is only supported for the following databases: SAP HANA, Sybase ASE, Oracle, Microsoft SQL Server, IBM DB4, IBM DB6) or the current table is of table class POOL (pool tables cannot be used for this option).

If you use this option, you need to define at least one key field as the grouping field. The field or list of fields are used to calculate a hash value between 0 and 255 which will be stored in the logging table. In order to optimize the distribution of data to the data transfer jobs, you need to define a key field which contains wide range of values (if most records for the key field contain the same entry then performance will not be improved). If the key field has too many records with the same value, you can also add additional key fields. As this impacts the performance of the calculation of the hash value, you should only define more key fields if really required.

In addition to the grouping field you also need to define the number of ranges. Each range will handle a specific set of values in field IUUC_REPL_GROUP so that the data for each range can be transferred at the same time. The number of ranges defines the maximum number of parallel jobs that can work on the same table. It does not define the real number of parallel process that will run on a table as this depends on the job settings (see section 2.2) and in which range the current changes happen. We recommend defining more ranges than the number of parallel jobs that should run on a table. But as the processing of each range also provides some overhead, the number of ranges should not be too high (a factor of 1,5 to- 2 is recommended, so if 5 jobs should run on a table, the number of ranges should be between 7 and 10).

You can change the number of ranges for a table for which the replication process is running as it only impacts the access plan calculation. If you are already using the option *Specify Ranges Manually*, and want to change the number of ranges, you need to reset the respective replication object using the processing step *Delete Load / Replication Objects*. Before the object is deleted, the table you must either suspend the replication process, or deactivate the configuration temporarily (both actions can be done by using the *SAP LT Replication Server Cockpit* (transaction LTRC).

3.2.2   Defining Table Ranges Manually

You can define table ranges manually for a table in order to improve the performance of the replication process.

You can do this in *Advanced Replication Settings* (transaction LTRS) under *Performance Options*. On the tab *Replication Options*, choose *Specify Ranges Manually (Expert)*.

You can define several ranges for a table, and for each range you can define several range conditions. You can use the range condition to define a range of values for every key field. If you have multiple value ranges for the same key field, you need to define separate ranges (as each key field can only be defined once in the range condition of one range).

The number of defined ranges also defines the maximum number of jobs that process the table. However, the final number jobs that process a table is defined by the job settings (see section 2.2). If you define range criteria in order to improve the performance of the replication process you need to ensure that every record fits to exactly one range.

Example: Ranges defined for table BKPS

You want to split the replication of table BKPF in mass transfer ID 001 into 4 ranges. You use field BUKRS to define the ranges:
Range 1: 0000 -> 2000
Range 2: 2001 -> 5000
Range 3: 5001 -> 8000
Range 4: 8001 -> 9999
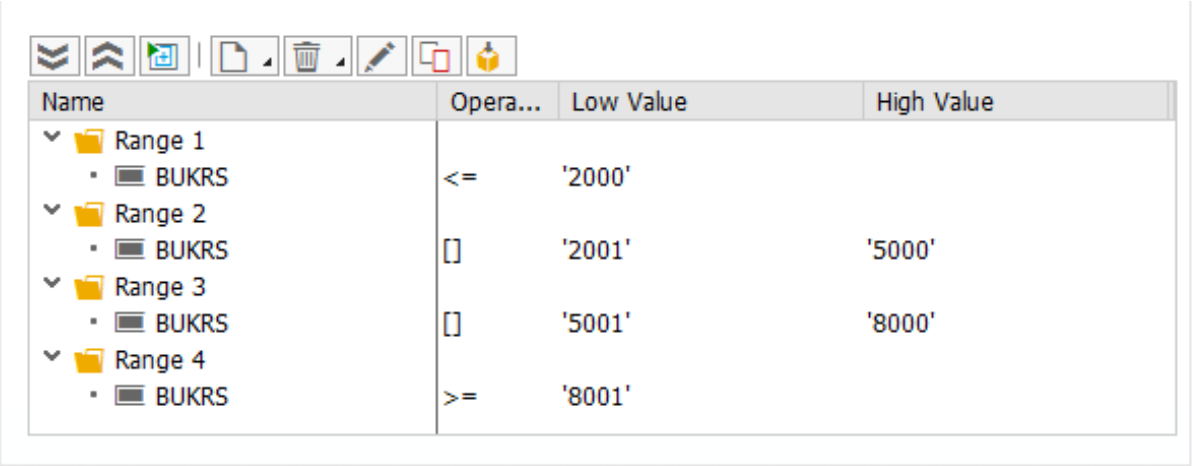
You define these filters in the *Advanced Replication Settings* (transaction LTRS) under *Performance Options* (on the *Replication Options* tab), as depicted in the following screenshot:

Name	Opera...	Low Value	High Value
⌄ 🗀 Range 1			
· ▦ BUKRS	<=	'2000'	
⌄ 🗀 Range 2			
· ▦ BUKRS	[]	'2001'	'5000'
⌄ 🗀 Range 3			
· ▦ BUKRS	[]	'5001'	'8000'
⌄ 🗀 Range 4			
· ▦ BUKRS	>=	'8001'	

Note

For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction LTRS.

Based on this information, SAP LT Replication Server uses the following ranges to select the data from the table:

Range 1: `SELECT FROM BKPF WHERE BUKRS <= '2000'.`

Range 2: `SELECT FROM BKPF WHERE BUKRS >= '2001' AND BUKRS <= '5000'.`

Range 3: `SELECT FROM BKPF WHERE BUKRS >= '5001' AND BUKRS <= '8000'.`

Range 4: `SELECT FROM BKPF WHERE BUKRS >= '8001'.`

4 Initial Load and Replication Process (Subscription-based Change Data Capture Mechanism)

You use the initial load to load data from the source system to the target system. You can do this in the SAP LT Replication Server Cockpit (transaction LTRC) by choosing *Data Provisioning*, and then *Start Load.* You can also use the option *Start Replication*, which starts the initial load and then begins the replication process. The replication process is used to keep data synchronized between the source system and the target system. You can trigger the replication process in the SAP LT Replication Server Cockpit (transaction LTRC) by choosing *Data Provisioning*, and then choosing *Start Replication*.

4.1 Reading Type

If the source system is an SAP S/4HANA system with release 2020 or higher, then the subscription-based change data capture mechanism is used to record changes. This recording mechanism only supports reading type 5 - Sender Queue. Only transparent tables or CDS view can be loaded or replicated.

4.1.1 Reading Type 5 - Sender Queue

Reading type 5, *Sender Queue* is for transparent tables and CDS views. The relevant data is extracted and temporarily stored in a compressed format in a queue in the sender system. The database optimizer decides on the most efficient way to extract the data. The sender queue requires additional space for the initial load (approximately 10% of the volume to be loaded).

The data in the sender queue is stored in table DMC_INDXCL. This table can grow significantly in size while an initial load is running. After the initial load is completed, the data is deleted from the table.

Example

You have the following data in your system:

KEY_1	KEY_2	KEY_3	DATA
100	2500	1000000120	Document 1
100	3750	1000000127	Document 8
100	8000	1000000123	Document 4
100	8000	1000000125	Document 6
200	2500	1000000122	Document 3
200	2500	1000000124	Document 5
200	8000	1000000121	Document 2
200	8000	1000000126	Document 7

The following table illustrates the effect of using reading type 5, *Sender Queue* for the sample data above:

Sender Queue (DMC_INDXCL) for Sample Data

CLUSTER ID	DATA
000001	0372DC93A1713B5F1AE50DAB…
000002	278D476A02E4EB60F527DD763…
000003	BDB52E893A9C0C27B5A277C3…
000004	5C8649E8822E5495D01B4A9A7…

Technical Settings

The default portion size for reading types 5 is 8 million bytes. As the portion size is calculated by multiplying the total bytes of all fields by the number of records in a portion, the actual portion size is typically smaller as not all fields are filled.

You can change the default portion size in *Advanced Replication Settings* (transaction LTRS) under *Performance Options* by using the field *Package Size*.

By default, a single access plan calculation job copies the data from the original table to the queue in the source system. The initial load is handled by multiple data transfer jobs.

You can further improve the process that copies the data from the original table to the queue in the source system by defining number of records that are processed by one access plan.

You can change the number of records in *Advanced Replication Settings* (transaction LTRS) under *Performance Options* by using the field *Number of Records*.

For example if there is a table with 108 million records and you specify a value of 20 million for the table, the system will generate 6 access plans which can run in parallel. The first 5 access plans will handle 20 million records each, while the last access plan will handle the remaining 8 million records. To define the value for

parameter *Number of Records*, you can divide the total number of records in the table by the number of access plan that you want to run in parallel.

The parameter *Number of Records* enables the parallel processing in the calculation phase but do not specify the number of parallel jobs that will be used. You can specify the number of calculation jobs in the *SAP LT Replication Server Cockpit* (transaction LTRC) on the tab *Administration*. By default, not more than 3 calculation jobs will process the same table. You can adjust the number of jobs that run in parallel for the same table by using *Advanced Replication Settings* (transaction LTRS). Under *Performance Options*, on the *Initial Load* tab, you can use the field *No. of Jobs* to specify the maximum number of jobs that will work on the table.

The parameter *Number of Records* must be greater than 50,000. However, we recommend using a minimum value of 10 million records.

4.2 Improving the Performance of the Initial Load

By default, up to three jobs will work on each table. You can adjust the number of jobs that run in parallel for the same table by using *Advanced Replication Settings* (transaction LTRS). Under *Performance Options*, on the tab *Initial Load Options* you can use the field *Number of Jobs* to specify the maximum number of jobs that will process the table. The parallel jobs settings in *Advanced Replication Settings* are used when initial load for the table is started. If you want to adjust the settings for a running initial load, you can do this in the *SAP LT Replication Server Cockpit*. On the *Participating Objects* tab, select the relevant table and choose *Edit Object Settings*. The relevant field is called *Number of Jobs*.

The actual number of jobs that process the same table also depends on the number of running data transfer jobs. So you might need to adjust the number of transfer jobs (and initial load jobs) if required. You can set the number of transfer jobs (or initial load jobs) in *Administration Data* tab in the *SAP LT Replication Server Cockpit* (transaction LTRC) in edit mode.

4.3 Improving the Performance of the Replication Process

By default, data replicated in one single process. To improve the performance of the replication process, you need to change the parameter *Number of Jobs* in transaction LTRS (on the tab *Replication Options*). If this parameter is changed after the replication process has started, you must delete the migration object and generate it again. Note that the migration object will be generated again automatically by the master controller job.

4.4 Filtering

In the SAP LT Replication system, in Advanced Replication Settings (transaction `LTRS`), you can filter the data that is transferred to the target system by using filter rules.

You can define filters which are used when selecting the data from the source system, to avoid the transfer of unnecessary data to the SAP LT Replication Server system. You can define such filters in the *Advanced Replication Settings* (transaction `LTRS`) under *Performance Options*. The filters you define are used for both the initial load and for the replication process.

On the tab *Filter Options*, you can define filters. You can define several filters and for each filter, you can define multiple filter conditions. For each filter condition, you can define a value range for every key field such that only specific values will be replicated to the target

If you need multiple value ranges for the same key field, you need to create a separate filter for each range (as each key field can only be used once in the filter condition of one filter). All filters are used when data is selected from the source system (using an AND operator), so only those records will be transferred that fit to all defined filter conditions.

In order for any additional settings to take effect, you must save your changes in the table before you start the initial load.

Example 1: Filter for one field for a single value

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load the records where the value of the field GJAHR is greater or equal to 2015. This filter is relevant for the initial load.

Note

> For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction `LTRS`.

Based on this information, SAP LT Replication Server uses the following WHERE clause to select the data from the table:

```
SELECT FROM BKPF WHERE GJAHR >= 2015.
```

Example 2: Filter for one field for multiple values

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load records where the value of the field BUKRS is equal 0010, or is between 1000 and 3000. This filter is only relevant for the initial load.

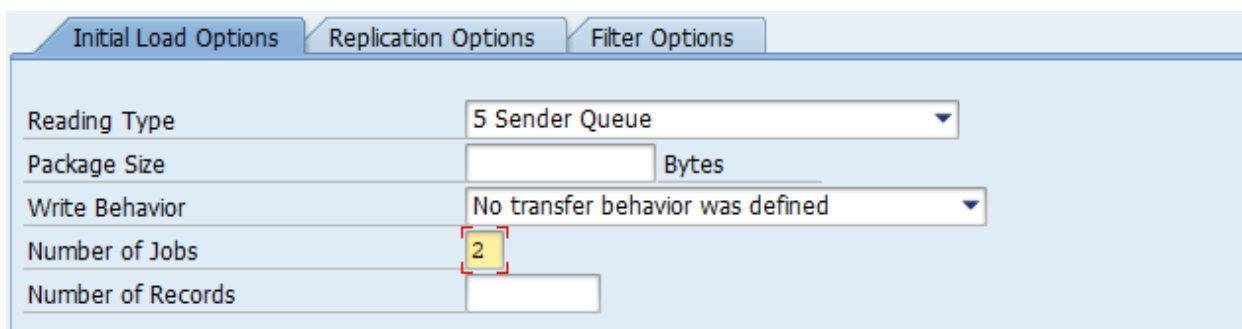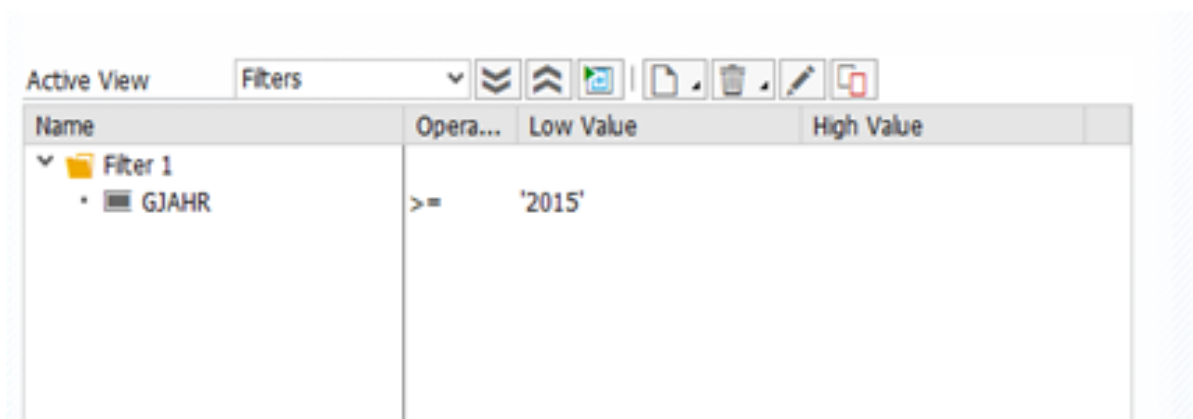| Active View	Filters	Opera...	Low Value	High Value
∨ 📁 Filter 1				
• 🖥 BUKRS		=	'0010'	
∨ 📁 Filter 2				
• 🖥 BUKRS		[]	'1000'	'3000'

Note

For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction `LTRS`.

Based on this information, SAP LT Replication Server uses the following WHERE clause to select the data from the table:

```
SELECT FROM BKPF WHERE BUKRS = '0010' OR BUKRS >= '1000' AND BUKRS <= '3000'.
```

Example 3: Filter for multiple fields

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load records where the value of the field GJAHR is greater or equal than 2015, and where the value of the field BUKRS is equal 0010 or is between 1000 and 3000. The filter is only relevant for the initial load.

Active View	Filters	Opera...	Low Value	High Value
∨ 📁 Filter 1				
∨ 🖥 BUKRS		=	'0010'	
• 🖥 GJAHR		>=	'2015'	
∨ 📁 Filter 2				
∨ 🖥 BUKRS		[]	'1000'	'3000'
• 🖥 GJAHR		>=	'2015'	

Note

For alphanumeric fields that only contain numeric values, you need to enter leading zeros. For example, consider a field with the data type VARCHAR(10) that only contains numeric values. If you want to filter for values greater than 10000, you need to enter value 0000010000 in transaction `LTRS`.

Based on this information, SAP LT Replication Server uses the following WHERE clause to select the data from the table:

```
SELECT FROM BKPF WHERE GJAHR >= 2015 AND BUKRS = '0010' OR BUKRS >= '1000' AND BUKRS
<= '3000'.
```

Example 4: Filter for one field using parallel jobs

You want to define a filter for table BKPF in mass transfer ID 001. You only want to load the records where the value of the field GJAHR is greater or equal to 2015. This filter is relevant for the initial load. As the number of relevant records is more than 200 million, you want to initial load the record with 2 paralleled jobs.

4.5 Processing of Logging Tables

In real-time replication mode, SAP LT Replication Server constantly checks all logging tables in the source system for new delta data. This means that all tables listed in the LT Replication Server Cockpit (transaction `LTRC`) that have the status *In Replication* are being processed by the data transfer jobs. As soon as all tables are processed, the jobs start processing all the tables again.

The default option is *Only Process Logging Tables with Records*. This means that SAP LT Replication Server identifies the logging tables that contain at least one record, and works on this list of reduced tables. If no changes occur in the source system, one replication job constantly checks for new updates, while all other data transfer jobs wait until the replication job signal that there is delta data to process.

However, if you most of the tables that have the status *In Replication* change frequently, you can change this option to the option *Process All Logging Tables*. Note that this can result in unnecessary resource consumption in the source system.

You can change this option in the LT Replication Server Cockpit. On the tab *Administration Data*, you can change the parameter *Options for Logging Tables*. If you change this option, you need to deactivate the configuration and activate it again afterwards in order for the change to take effect.

www.sap.com/contactsap