

System Copy GT

1.8 — Last update: 2016/05/11

Basis Technologies

Table of Contents

- Introduction 1
- Prerequisites 2
- Executing System Copy GT 3
 - Program Parameters / Selection Screen 4
 - Technical Settings..... 6
 - Advanced Options..... 8
 - Execution Sequence 11
 - Performance Considerations 12
 - Monitoring Program Execution 13
- Conversion Logs 15
- Troubleshooting 18
 - Tuning System Copy GT 19
 - Error Handling..... 20
- Important Notes 21
- Support from Basis Technologies 22

Introduction

The purpose of this document is to provide end user documentation on System Copy GT. This is an accelerated version of the standard SAP transaction BDLS. This is a transaction entitled “Tool: Conversion of Logical System Names” the purpose of which is to update all database tables that contain “logical system” fields. The most common use of this transaction is as a part of the system copy process, performed by the SAP Basis administration team. Once either a database copy or client copy has been performed from the source system (e.g. production) into the target system (e.g. pre-production / test) then this transaction must be run to adapt the runtime environment to the current state of the data. The type of ABAP stack system (e.g. ECC, BW) will determine how many and which tables must be updated. Typically, for an ECC/ERP system there are in excess of 1500 tables that must be updated. With the standard SAP BDLS transaction, this process can take anywhere from 1 hour to 5 days dependent upon the amount of data within that system. System Copy GT enables a significantly accelerated version to execute this process much faster. Powered by the Node5 Diffuser, it uses multiple processors to achieve this acceleration. Furthermore, it allows to execute multiple system name conversions in a single run.

This document describes pre-requisites and additional parameters required to execute System Copy GT.

Prerequisites

Just like the standard BDLS transaction, no other activities should be performed in any part of the system including communication with other systems. Equally, all IDocs in the system should have been processed. You must run this program within each client that requires it as the logical system ID usually contains the client number.

Some additional recommendation prior to execution are:

1. **Change Operation Mode:** The Node5 Diffuser and hence System Copy GT use background processes to perform the parallel logical system conversion. Please increase the number of background work-processes that are available in the SAP system prior to execution (appropriate to the size of the system)
2. **Redo Log Backups:** It is recommended to make sure there is enough space for redo logs in the targeted system
3. **Multiple system name conversions:** In case of multiple logical system name conversion it is ideal to exclude big tables which are not relevant to the additional name conversions. this can help performance significantly, but it is by no means mandatory
4. **Oracle DB:** If your system uses an Oracle database please make sure that there is enough tablespace available prior to the system conversion
5. **BDLS Notes:** Please make sure to apply any OSS notes you already use to run standard BDLS. System Copy GT makes calls to SAP standard code and this will ensure that we avoid possible issues in these cases
6. **Disable archive logs:** It is recommended to disable archive logs, the generation of logs can significantly slow the database update down
7. **Use Server Groups:** Jobs scheduled within a server group are given higher priority (job class A). In addition, if the specified server group contains an application server that runs on the database the jobs are planned there, this reduces network load significantly

You must also have installed the Node5 Architecture as well as System Copy GT in that order. These will have been provided by BTI in the form of a transport request file. The only mandatory requirement is to then have the license key installed for the given SAP system. This will be provided by BTI as well.

Executing System Copy GT

System Copy GT is executed under transaction /BTR/FBDLS. Please note that if entering this in the transaction area, you must use a leading “/n” – i.e. “/N/BTR/FBDLS”. This transaction references program /BTR/MDR_PP_FBDLS_MULTI.

You must have appropriate authorizations in place as per standard BDLS. This includes authorization object “B_ALE_LSYS”.



Program Parameters / Selection Screen

Once you have executed transaction /N/BTR/FBDLS, you will be prompted with the first part of the selection screen that is very similar to the standard BDLS transaction as below.



Note the program should always be run in background to ensure that the loss of the dialog session does not cause a problem, monitoring is easily achieved through the /N/BTR/MINICUBE transaction.

System Copy GT: Fast Conversion of Logical System Names

  Technical Settings

Conversion of logical system names

Old Logical System Name

New Logical System Name ☒

☒ Conversion of Client-Dependent and Client-Independent Tables
(e.g. Renaming Original System or Following Database Copy)

☐ Conversion of Client-Dependent Tables
(e.g. Following a Client Copy)

☒ Test run ☒ Check Existence of New Names in Tables



Number of Entries per Commit

Tables to be Converted to

However, System Copy GT has enhanced features from the standard transaction.

- Technical Settings – When you select this button, the options for using Node5 Diffuser are displayed. This will be described in the next section, [Technical Settings](#)

System Copy GT: Fast Conversion of Logical System Names

  Technical Settings

- **Advanced Options** – The selection screen includes a number of options to aid performance and tailor the running of System Copy GT to your needs, details on each feature are described in the [Advanced Options](#).

System Copy GT: Advanced Options

☒ Single commit size
☐ Multiple commit sizes ->Convert disproportionately larger tables more efficiently

☒ Single name conversion
☐ Multiple name conversion ->Convert more than one logical system name in a single run

Additional Options

☐ Ignore check on old LS value
☒ Process special objects
☒ Randomize intervals
☐ Dynamic Interval Keys
☐ Ignore contents in bdsexz
☐ Postpone position log update

Parallel interval generation

☒ Generate intervals in parallel
Number jobs for interval gen

Everything else in the selection screen work in the same way as standard BDLS. For details on these parameters, please refer to the standard SAP documentation.

Technical Settings

On pressing the “Technical Settings” button a screen is displayed as below.

System Copy GT: Fast Conversion of Logical System Names

Instance Settings

Label

Interval Settings

Interval Count

Interval Size

Distribution

☒ Number of batch jobs across all servers

☐ Distribution according to server group

☐ Manual Distribution

☐ Run online as a single process (debugging mode)

Other settings

☐ Wait for run to complete ☐ Launch Transformation Program after completed run

Distribution List

Message log level

The most important parameters for System Copy GT are:

- Number of batch jobs across all servers – Specify the number of jobs with which you want to run the program with. Ensure that sufficient background work processes are available. Be mindful that too much parallel processing can cause database locking issues when updating the same table. Please refer to the [Performance Considerations](#) section later in this document for further information.
- Interval Size – This parameter defaults to the standard BDLS parameter “Number of Entries per Commit” and represents the same purpose. It is critical in the sense that the number of entries can directly affect performance of both the standard BDLS transaction and this MDR Process Pack. If this value is set to too low, then far too many COMMIT's are made to the database – introducing an unnecessary overhead. In addition, too many intervals will be created which will also introduce an unnecessary overhead. Conversely, if set too high the commit area can become very large and even

cause instability with database interruptions. The optimal value to specify here is in the range of 50,000 to 150,000. It is suggested to set this to a value of 100,000, which is the default setting.

An explanation for the function of each field on the “Technical Settings” screen is as follows:

- Label – The first is a label that can be specified to identify this particular execution
- Interval Count –This specifies the number of intervals (chunks) that the total amount of work to be done is to be broken up into, this is not relevant for System Copy GT
- Interval Size –This specifies the “number of objects” to be put into each interval to be then worked upon independently
- Number of batch jobs across all servers –This specifies the number of processes with which to run the Diffuser program
- Distribution according to server grouping –This allows the distribution of jobs over one server group to control the number of processors available to this Diffuser instance
- Manual Distribution –The server grouping above can also be distributed manually
- Run online as a single process (debugging mode) –This is only used when debugging Diffuser programs and ensures the whole program runs sequentially
- Wait for run to complete before finishing –This is often used when running Diffuser programs on-line or when executing them via a job scheduler. It will ensure the parent process waits until all child processes have completed. Once all child processes have finished, control is returned to the parent for completion
- Distribution List –After a Diffuser program completes it is able to send a SAP office document or external email to a set of recipients that can be specified here
- Message log level – Lower limit for the priority of messages output to the application log. For example, you can restrict output of informational application log messages by increasing the log level via this parameter




Please Note: All of these parameters can be defaulted either when executing the transaction on-line or saved within a variant if executed in the background. See the Node5 Architecture Guide [Default Technical Settings](#) for information on how to setup defaults.

Advanced Options

The advanced options available for System Copy GT are:










Multiple commit sizes – This functionality allows to specify multiple numbers of entries per commit for selected tables. In some cases creating smaller intervals only for selected tables can help improve overall performance without massively increasing the total number of intervals to be processed in a program run. See example below:

☐ Single commit size
☒ Multiple commit sizes ->Convert disproportionately larger tables more efficiently

Number of entries per commit for selected tables				
Number of entries per commit	<input type="text" value="50,000"/>	Dynamic Key	<input type="checkbox"/>	Tables <input type="text" value="VBAP"/> 
Number of entries per commit	<input type="text"/>	Dynamic Key	<input type="checkbox"/>	Tables <input type="text"/> 
Number of entries per commit	<input type="text"/>	Dynamic Key	<input type="checkbox"/>	Tables <input type="text"/> 

Multiple name conversion – This option allows to include additional logical system name conversion in one single run. In addition to time performance improvement it is very convenient from a usability perspective. Furthermore, specific table list can be assigned to each individual additional conversion to improve performance. By doing this unnecessary update requests to the database can be avoided. See example below:

☐ Single name conversion
☒ Multiple name conversion ->Convert more than one logical system name in a single run




Additional Conversions				
Additional Conversion	Old <input type="text" value="BWPCCLNT100"/>	New <input type="text" value="BWQCLNT100"/>	Tables	<input type="text" value="FAGLFLEXA"/> 
Additional Conversion	Old <input type="text" value="CRPCCLNT100"/>	New <input type="text" value="CRQCLNT100"/>	Tables	<input type="text" value="VBRP"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 
Additional Conversion	Old <input type="text"/>	New <input type="text"/>	Tables	<input type="text"/> 

A number of additional options are also available as checkboxes, the details of each of these are described in more detail below the screenshot.

Additional Options	
<input type="checkbox"/>	Ignore check on old LS value
<input checked="" type="checkbox"/>	Process special objects
<input checked="" type="checkbox"/>	Randomize intervals
<input type="checkbox"/>	Dynamic Interval Keys
<input type="checkbox"/>	Ignore contents in bdsexz
<input type="checkbox"/>	Postpone position log update

- **Ignore check on old LS value** – The standard operation of both BDLS and System Copy GT is to only update logical system fields with the specified input parameter “Old Logical System Name”. By selecting this checkbox, it overrides the restriction on the old logical system value and updates ALL records, regardless of their current value. Please ensure that this parameter is used appropriately. The default value is to turn this off. It is useful when previous BDLS executions have not been fully executed in the past.
- **Process special objects** – Prior to updating the logical system names of all relevant tables, some special tables must be updated. These include tables such as SOOD (SAP Office Documents). Equally, after all tables have been updated, some additional tables are updated. This parameter ensures that the standard BDLS process for updating these special objects is also performed. The default for this parameter is turned on.
- **Randomize intervals** – When updating tables through the use of parallel processing, locking of database rows is required during the commit scope. This is a function of the database and not the application. When the same table is updated concurrently, it can lead to instability if too many processes are updating the same table. To mitigate this restriction, the intervals (tables to be updated) is randomized so that the same table is not constantly updated “together” and instead are randomly spread out amongst the total interval set. This parameter is by default set to true and we recommend this is not changed.
- **Dynamic Interval Key**– System Copy GT offers two ways of interval generation. The first one is a simple and efficient algorithm which determines a number of up to three keys to break up tables by once at the beginning of interval generation for every table. This algorithm is set by default and applies when option Dynamic Interval Key is unchecked.
If Dynamic Interval Key option is checked a second algorithm is applied for the interval generation instead. Dynamic Interval Key Generation works by determining the best number of keys by which to break up a table for every interval. This provides a more precise interval key and therefore quicker database access during the actual table update process of System Copy GT. The drawback is that the interval generation can take longer than the default algorithm.
In general, it is recommended to use the default algorithm for most tables and only use Dynamic

Interval Key Generation for disproportionately big tables or tables with long keys (e.g. 4 or more key fields). This can be specified in the Multiple Commit Sizes option as shown below:

Number of entries per commit for selected tables				
Number of entries per commit	<input type="text" value="50,000"/>	Dynamic Key <input checked="" type="checkbox"/>	Tables	<input type="text" value="FAGLFLEXA"/> 
Number of entries per commit	<input type="text"/>	Dynamic Key <input type="checkbox"/>	Tables	<input type="text"/> 
Number of entries per commit	<input type="text"/>	Dynamic Key <input type="checkbox"/>	Tables	<input type="text"/> 

- Ignore contents in bdlsexz – In certain situations table bdlsexz may be used concurrently by the standard tool. To avoid conflict you can ignore the contents of this config table when you run System Copy GT
- Postpone position log update – This option allows

The final section deals with Interval Generation, which is the process of breaking up all the tables involved with the BDLS run into the commit sizes.

Parallel interval generation	
<input checked="" type="checkbox"/> Generate intervals in parallel	
Number jobs for interval gen	<input type="text" value="10"/>

- Parallel interval generation – Prior to the actual conversion step System Copy GT breaks up the tables to be converted into ranges of rows (intervals) that can be updated independently. Any one table can only be broken up sequentially but the same process can be conducted on multiple tables in parallel. Selecting this option can significantly improve overall performance.
- Number of jobs for interval – This determines how many parallel processes are allocated for this purpose



Advanced option values are defaulted when you execute the program and the defaults are generally what you should execute with.

Execution Sequence

After you have entered the appropriate parameters for System Copy GT, the job should be started as a background job with a variant. The execution sequence is as follows:

1. Creation of “intervals” that represent the work to be processed
2. Processing of “special objects” of type A (prior to conversion of core tables)
3. Main parallel processing of updates to logical system fields
4. Processing of “special objects” of types TXZ (after conversion of core tables)
5. Collation of results for viewing of what has been done
6. Display of output (ad-hoc)

Steps 1 and 3 are able to be completed using the Node5 Architecture’s Diffuser to utilize multiple processors. The remainder of the steps in the above sequence are done in a sequential manner with a single work process.

Performance Considerations

System Copy GT is using multiple processors to update the logical system names considerably faster. It is also not hanging on to database cursors which can often slow down the existing standard transaction, as each update to the database is done independently of each other. Too many concurrent updates to the same database table could trigger database deadlocks. To avoid this as much as possible, System Copy GT randomizes the sequence of database table update requests. However, it is still possible that concurrent updates occur and this is acceptable up to a certain number of parallel processes.

The actual maximum amount of processors you can use on your system is dependent upon the capacity of the underlying database. It is recommended that you start the transaction with a smaller number of processes (e.g. 4-5) and then gradually increase the processing capacity if required. Please refer to the following section on how to achieve this.

Monitoring Program Execution

After the program has been set running, you are able to immediately monitor the program execution. Alternatively, in another session, you are able to go to the transaction /BTR/MINICUBE check the date and user are correct for the run you are searching for.

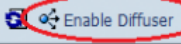
MiniCube - Explorer

Select Options

Instance ID	<input type="text"/>	to	<input type="text"/>	
Started By	USER1	to	<input type="text"/>	
Start Date	28.02.2015	to	<input type="text"/>	
Start Time	00:00:00	to	00:00:00	
End Date	<input type="text"/>	to	<input type="text"/>	
End Time	00:00:00	to	00:00:00	
Instance Status	<input type="text"/>	to	<input type="text"/>	
Diffuser Program	<input type="text"/>	to	<input type="text"/>	

You should see that it is currently in progress and the percentage complete, by pressing the Diffuser Mode button more options are revealed.

MiniCube - Explorer

 Enable Diffuser

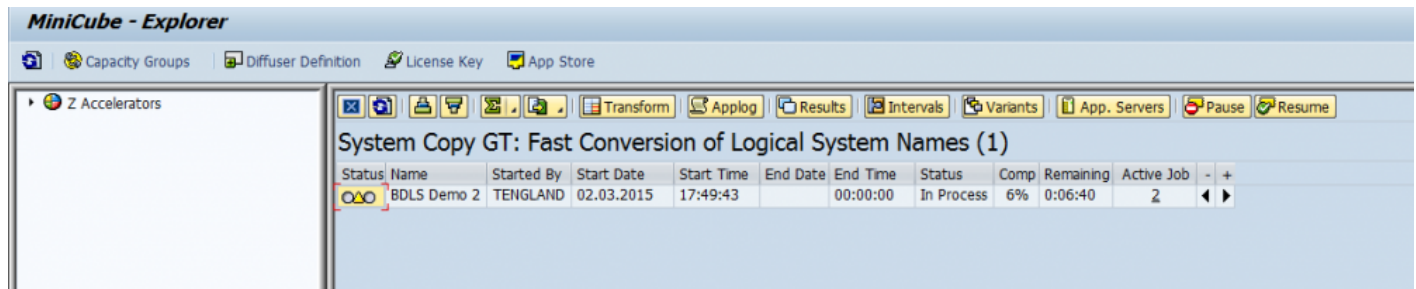
► Z Accelerators

System Copy GT: Fast Conversion of Logical System Names (1)

Status	Name	Started By	Start Date	Start Time	End Date	End Time	Status	Comp	Remaining
	BDLS Demo 2	TENGLAND	02.03.2015	17:49:43		00:00:00	In Process	0%	0:01:47

You are now able to see the number of jobs working on the run and able to increase (or decrease) the number of parallel jobs being run by selecting the “left” and “right” arrows appropriately, or even pause and restart the run if required. You are also able to select the instance and then choose “Intervals” to see which

intervals have currently been processed, which are currently in progress and those that are yet to be done. Select the “Refresh” button to refresh where the intervals are currently.

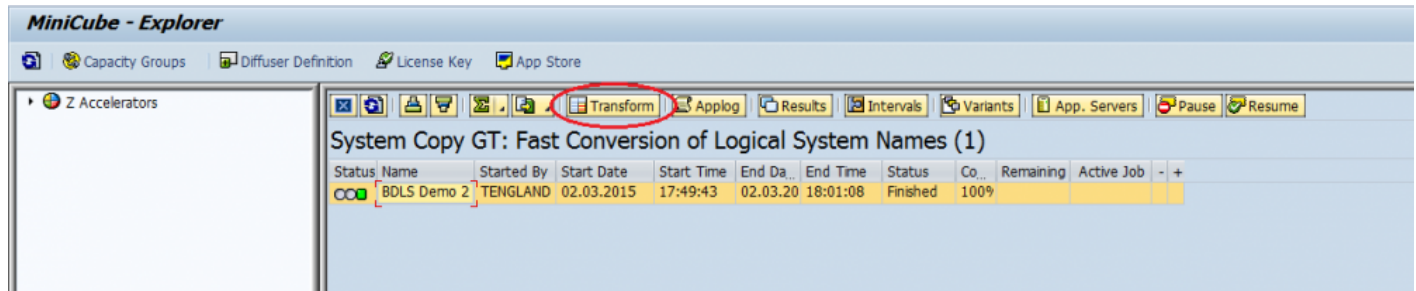


It is recommended that during the run, sometime is spent monitoring through transaction SM66, to find the optimum number of processors, this needs to be done after the interval creation has completed. If the commit times there are under 100 seconds then increasing 5 processors and then monitoring again is a good option. If the runtimes rise over 100 seconds it is advisable to reduce the number of processors. Checking that the database load may also give another indication on the optimum number of processors.

For further information on running, monitoring and administering Diffuser programs, please refer to the [Node5 Administrators Guide](#).

Conversion Logs


Similar to standard BDLS System Copy GT displays conversion logs and counts at the end of every system update. To access the logs select the instance in the MiniCube list and hit “Transform” on the menu bar.



The conversion log is then displayed as below.

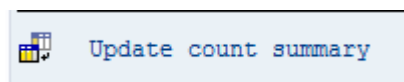
System Copy GT: Conversion Logs		
System Copy GT: Conversion Logs		
Update count summary		
System Copy GT: Conversion Logs		
Log for test run: MDRPOC_1 -> MDRPOC_2 in system (initiator)		
		02.05.2013 20:39:55 COLIVER
New entry MDRPOC_2 has been entered in field LOGSYS of table TBDLS		
New entry MDRPOC_2 has been entered in field LOGSYS of table TBDLST		
Table T000 was not relevant for conversion		
Table Name	Field Name	Number of Relevant Entries
/BEV3/CH1030BKPF	ANSYS	0
/BEV3/CHFIA	LOGSYS	0
/BEV3/CHFIP	LOGSYS	0
/BEV3/CHFIT	LOGSYS	0
/BEV3/CHVGRA	LOGSYS	0
/BEV3/CHVGRP	LOGSYS	0
/BEV3/CHVGRT	LOGSYS	0
/BI0/PTCTSYSID*	LOGSYS	0
/BI0/SLOGSYS*	LOGSYS	0
/BTI/MDE_CRM_T06	LOGICAL_SYSTEM	0
/BTR/MDR_TBICO	RECLOGSYS	0
/CCIS/PBU18A	LOGSYS	0
/CCIS/PBU18C	LOGSYS	0
/CCIS/PBU18P	LOGSYS	0
/CCIS/PBU18T	LOGSYS	0

In case of a multiple name conversions these will be shown below the main conversion in the same format as well.

System Copy GT: Conversion Logs		
		
System Copy GT: Conversion Logs		
Log for test run: BD1CLNT100 -> BD1CLNT100 in system (initiator)		
02.03.2015 17:49:43 TENGLAND		
Table Name	Field Name	Number of Relevant Entries
/BEV3/CH1030BKPF	AWSYS	0
/BEV3/CHFIA	LOGSYS	0
/BEV3/CHFIP	LOGSYS	0
/BEV3/CHFIT	LOGSYS	0
/BEV3/CHVGRA	LOGSYS	0
/BEV3/CHVGRP	LOGSYS	0
/BEV3/CHVGRT	LOGSYS	0
/BI0/PTCTSYSID*	LOGSYS	0
/BI0/SLOGSYS*	LOGSYS	0
/BTI/MDE_CRM_T06	LOGICAL_SYSTEM	0
/BTR/MDR_TBICO	RECLOGSYS	0
/CCIS/PBU18A	LOGSYS	0
/CCIS/PBU18C	LOGSYS	0
/CCIS/PBU18P	LOGSYS	0
/CCIS/PBU18T	LOGSYS	0
/CCIS/PROFTAX01	LOGSYS	0
/CCIS/PROFTAXA	LOGSYS	0
/CCIS/PROFTAXP	LOGSYS	0
/CCIS/PROFTAXT	LOGSYS	0
/EACC/TLEAD_BW	LOGSYS	0
/ISDFPS/ALE_OBJ	LOGSYS	0
/ISDFPS/ALE_RBDS	LOGSYS	0
/ISDFPS/ALE_STAT*	LOGSYS	0
/ISDFPS/ALE_SYS*	CENTRAL_SYS	0
	CUA_SYS	0
	LOGSYS	0



The button below in the results can be used to see a summary of the results.



Summary results appear as below.

System Copy GT: Conversion Logs				
Table Name	Field Name	Log.System	Σ	Updates
BKPF	AWSYS	MDRPOC_2		3.836.000
BKPF				■ 3.836.00
				■ ■ 3.836.0

Troubleshooting

Tuning System Copy GT

Once a run of System Copy GT has been completed then the results can assist in tuning your next run. Using the Intervals button in Diffuser Mode enables you to see all of the intervals and their runtime. By sorting the runtimes you are able to see which intervals took the longest, if consistently the same table is seen as having the slowest interval runtimes then you may be able to take steps to tune the next run.

Consider the following options:

- Does this table have entries that need to be updated? Can it be excluded?
- Could the commit size be reduced, see the [Advanced Options](#) section
- Does the table have a complex key involved? Should you consider the “Dynamic Key” option?

A run with intervals exposed and runtimes sorted is shown as below.

System Copy GT: Fast Conversion of Logical System Names (1)

Status	Name	Started By	Start Date	Start Time	End Date	End Time	Status	Co...	Remaining	Active Job	-	+
Finished	BDLS Demo 2	TENGLAND	02.03.2015	17:49:43	02.03.	18:01:08	Finished	100%				

BDLS Demo 2 (1931)

Status	Inter..	Low Value	High Value	Status	Runti..	Results
Completed	54	NZTMP_3US2100	000261436 LOGSYS	Completed	0:03:34	2
Completed	368	NZTMP_3US2100	000261475 LOGSYS	Completed	0:01:34	2
Completed	414	NZTMP_3US2100	000261475 LOGSYS	Completed	0:00:35	2
Completed	1758	NTBTCTO	0002614366 RECLOGSYS	Completed	0:00:16	2
Completed	920	NBKPF	0002614561 AWSYS	Completed	0:00:06	2
Completed	1541	NSWWWIHEAD	000261436 REMOTE_SYS	Completed		2
Completed	30	NBKPF	0002614489 AWSYS	Completed	0:00:05	2
Completed	214	NBKPF	0002614673 AWSYS	Completed		2
Completed	261	NBKPF	0002614557 AWSYS	Completed		2
Completed	1209	NBKPF	0002614536 AWSYS	Completed		2
Completed	1486	NBKPF	0002614407 AWSYS	Completed		2
Completed	46	NBKPF	0002614476 AWSYS	Completed	0:00:04	2
Completed	231	NBKPF	0002614484 AWSYS	Completed		2
Completed	352	NBKPF	0002614713 AWSYS	Completed		2
Completed	387	NBKPF	0002614469 AWSYS	Completed		2
Completed	464	NTBTCTO	0002614367 RECLOGSYS	Completed		2
Completed	621	NBKPF	0002614461 AWSYS	Completed		2
Completed	1456	NBKPF	0002614527 AWSYS	Completed		2

Error Handling

In the unlikely event of finding an interval ending in error the whole run will be in error, because System Copy GT uses the Diffuser Framework it is very easy to reprocess the error, the interval can simply be rerun, meaning you don't have to execute the whole run again or even run it for the table that failed. For details see the [Reprocess Error](#) section of the administrators guide.

Important Notes

Please keep in mind the following when executing System Copy GT:

- Increase available background jobs – The Node5 Architecture makes use of background work processes rather than dialog. Increase how many are available in the system using operation modes to ensure there is sufficient capacity available for multiple processors
- Set the commit interval appropriately – The COMMIT interval is suggested based upon the underlying SAP database type. The default value is 100,000. The recommended range is from 50,000 to 150,000
- Monitoring Options – The progress of the program can be monitored through the use of the standard Node5 MiniCube monitoring tool to understand progress. This is described in the [Monitoring Program Execution](#) section.

Support from Basis Technologies

Raising Support Tickets

To request support from Basis Technologies on any issue relating to our product sets (Transport Expresso, DevOps, Diffuser or Utilities) , a ticket should be raised via the following email address:

support@basistechnologies.com

Sending an email to this address will automatically create a ticket in Zendesk, the ticketing tool used by Basis Technologies.

Please include as much information as possible about the issue (product, version, error messages, steps to replicate, screenshot attachments) in the email. In addition, please also include your own contact details in your email.

Please reflect any high priority issues by including URGENT or HIGH PRIORITY at the start of the email subject.

Support Escalation

If you have any concerns with the service you are getting from Basis Technologies support, or wish to escalate any high priority issues please email **supportescalation@basistechnologies.com**

Require additional Information or Services?

If additional information or services relating to any of Basis Technologies product sets is required, you can contact us via the above support@basistechnologies.com address, or alternatively by contacting your assigned Basis Technologies Account Director.