Node5 Administrators Guide

8.0 — Last update: 2016/05/11

Basis Technologies

Table of Contents

Audience **2**

Introduction **3**

Basic Concepts **4**

- Node5 Architecture 5
- Node5 Diffuser 6
- Node5 MiniCube 7
- Instance 8
- Intervals 9
- Capacity Groups 11

Maintaining Licenses **12**

- Installing a License 13
- Check installed keys 15
- List installed products 16

When to use Diffuser **18**

Setting up a Diffuser program **19**

- Program Definition 20
- Default Technical Settings 22
- Interval Generation 24

Running Diffuser Programs **26**

- Technical Settings 27

Accessing MiniCubes **31**

Administering Diffuser Programs **33**

- Diffuser Mode 34
- Intervals 35
- Results 36
- Variants 37
- App Servers 38
- Increase or Decrease Jobs 39
- Pause 40
- Resume 42
- Delete 43
- Force Error 44

Reprocess Error 45
Debug an Interval 48
Rename Instance 50

Scheduling Diffuser Programs 52

Job Distribution 53
Server Group Distribution 54
Manual Distribution 55

Maintenance of MiniCube results 56

Performance considerations 57

Security Enhancements 58

Software Support 59
Online Forum 60
Support from Basis Technologies 61

Audience

This guide has been developed for the following audience:

- SAP® Basis Administrators
- SAP® performance specialists or advisers

Introduction

Basis Technologies' Node5 Architecture provides a framework and run-time environment for custom developed and standard SAP programs that must work with large volumes of data. It provides for dynamic execution such that these programs can run within acceptable time constraints to deliver timely business information. This is achieved by:

- Providing a framework and methodology for development of reports such that they can be scaled according to the resources available in order to maximize use of the available hardware
- Providing a framework for execution of ABAP code using parallel processing such that report run faster and more efficiently
- Enabling huge volumes of data to be processed within dramatically reduced timescales
- Providing a framework toolset for the monitoring and administration of the parallel execution of reports
- Facilitating the presentation and delivery of reports, enabling the business to interactively access current and historical report information

Basic Concepts

The basic concepts of operating the Node5 Architecture are shown as below:

- Node5 Architecture
- Node5 Diffuser
- Node5 MiniCube
- Instance
- Intervals

Node5 Architecture

The Node5 Architecture is the core of all of Basis Technologies software solutions.

For a program to be accelerated by the Node5 Diffuser, it can either be developed as a custom Z Accelerator or provided as a prepackaged program supplied by Basis Technologies (as a GT, GTi or BDi App). The key features to accelerate a program are the Node5 Diffuser and Node5 MiniCube.

Node5 Diffuser

The Node 5 Diffuser takes a large data set and splits it up into small pieces of data called Intervals these can then be processed with multiple processors running. The Node 5 Diffuser allows the number of processors to be increased or decreased at runtime, including pausing and restarting a run. A group of processors can be assigned to a Capacity Group to allow processing power to be dynamically distributed across programs.

This can be utilised by custom programs as a Z Accelerator or by programs supplied prepackaged by Basis Technologies (the GT, GTi and BDi Apps).

Node5 MiniCube

When a Node5 Diffuser run completes it can store the results as a Node 5 MiniCube inside the Node5 Architecture, these can be retrieved and further selections applied to this data. The data can also be supplied in an interactive way so that for example ALV or drilldown features can be used.

The transaction /BTR/MINICUBE also allows the monitoring of runs and with the Diffuser Mode the instance resources can be increased at runtime, see Administering Diffuser Programs.

Instance

Each run of a program using the Node5 Diffuser is called an ‘Instance’. This can be given a separate label and, using the framework, a user can view a number of previous instances and pick which they wish to view the details of. The framework allows the saving of data against the instance.

Intervals

The Node5 Diffuser works on the principal that data processing can be divided into independent “pieces” of work. These pieces of work are referred to as Intervals and usually represent a range of master or transactional data that needs to be worked upon. This, for example, might be Intervals of Business Partner Numbers, Sales Order Numbers, or Account Numbers. The Node5 Diffuser ensures that these Intervals of data are processed using multiple processors instead of the traditional sequential approach with one processor.
An interval can be thought of as a range of values that represent X number of master or transaction data objects. It is based on some data domain having a data type and a length. An example interval might be “All Sales Order’s from 1000 through to 2000”. It can then be implied that an interval has a Low value (e.g. 1000) and a High value (e.g. 2000).
If an SAP system has 100,000 Sales Orders, and an ABAP report is required to process all of them, then this range of Sales Orders can be broken down into, for example, 100 intervals, each representing 1000 Sales Orders. The list of 100 intervals might look something like:

Interval	Low	High
1	1	1000
2	1001	2000
3	2001	3000
…	…	…
100	99001	100000

The concept of an Interval Object is used to create these Intervals for use by a Diffuser program this is where an object such as a Sales Orders are broken down before the Diffuser program is run. As part of the definition of a Diffuser program, an Interval Object is selected. This refers directly to the type of Intervals the program uses, for example, some Interval Objects that are provided with Diffuser are:

- Sales Order
- Business Partner
- Contract Account

The generation of Interval Objects is detailed in the Setting up a Diffuser program Interval Generation section.

When using Z Accelerators intervals can also be built at runtime. This is programmed into the Diffuser program and is useful where you have complex intervals or need to define the number of intervals down to a smaller size; a subset of materials, for example. This concept is detailed in the Z Accelerators – Advanced Concepts – Dynamic Interval Generation section.

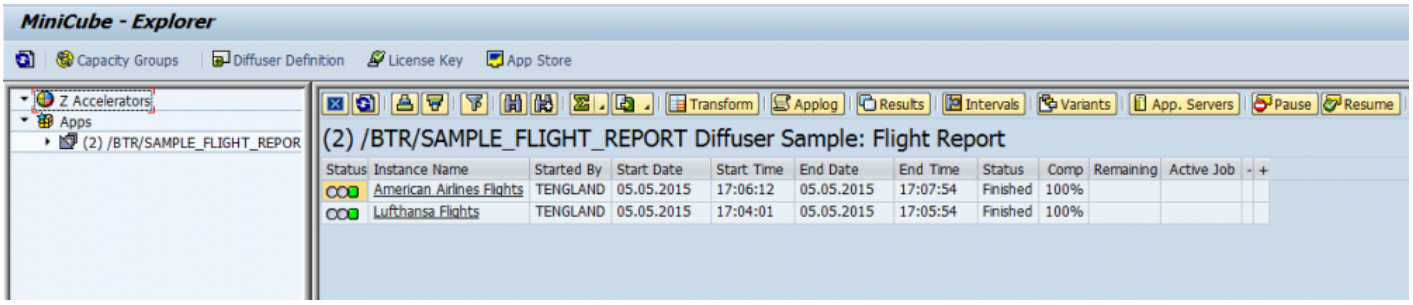
Capacity Groups

Capacity Groups is a powerful tool of the Node5 Architecture created to enhance its system resource administration capabilities. While the Node5 Diffuser provides a parallel processing platform and runtime environment which makes the execution of ABAP code faster and more efficient, Capacity Groups offer an advanced administration framework for the consumption of system resources by Diffuser enabled programs. The tool determines how many background processes a Diffuser program can use on one or more selected servers, on a specific time pattern, and its relative priority to other Diffuser programs running at that point.

Maintaining Licenses

The Transaction /N/BTR/LICENSE is used to access the License Administration for Diffuser.

It can also be accessed via the MiniCube transaction /N/BTR/MINICUBE after pressing the Diffuser Mode button and selecting the license key button.



The key components to managing your license are as below.

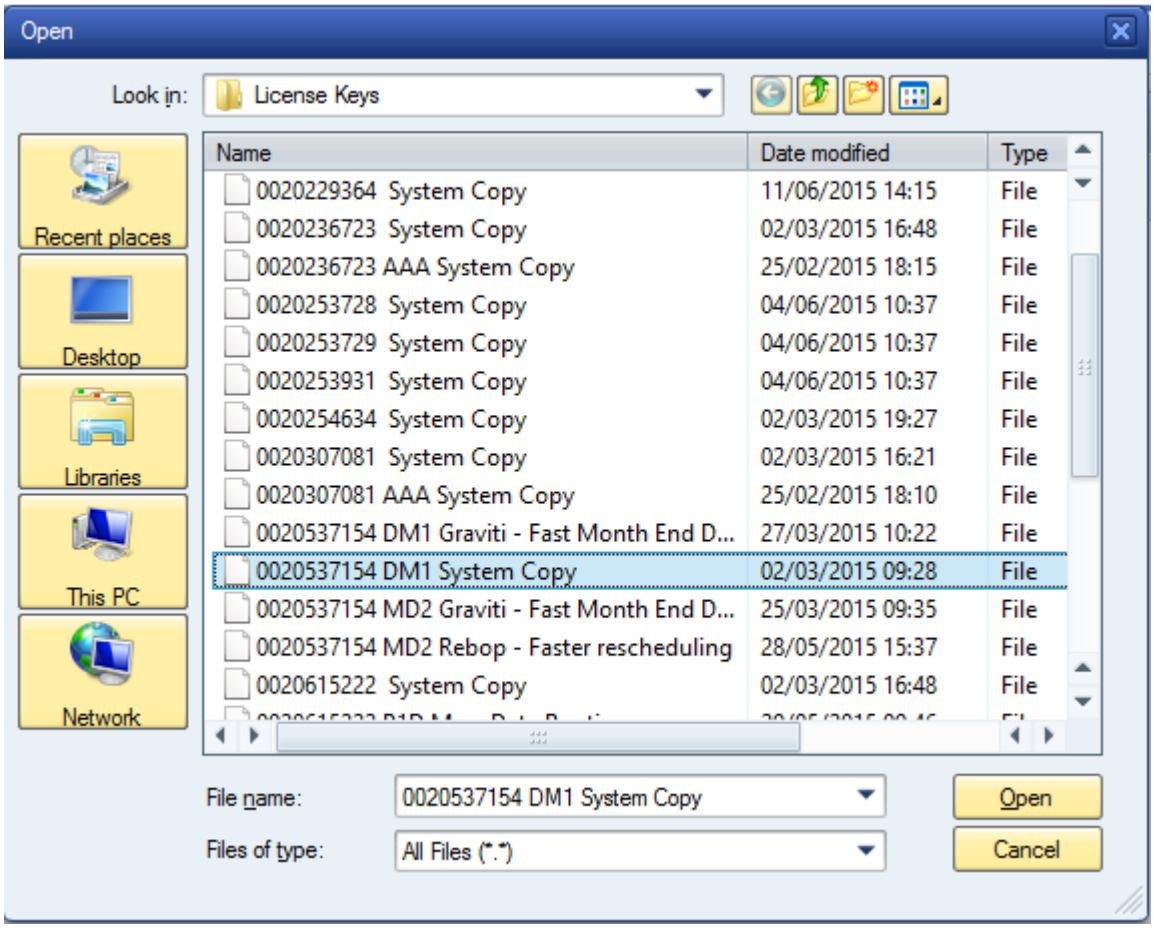
- Installing a License
- Checking installed Licenses
- Check installed keys

Installing a License

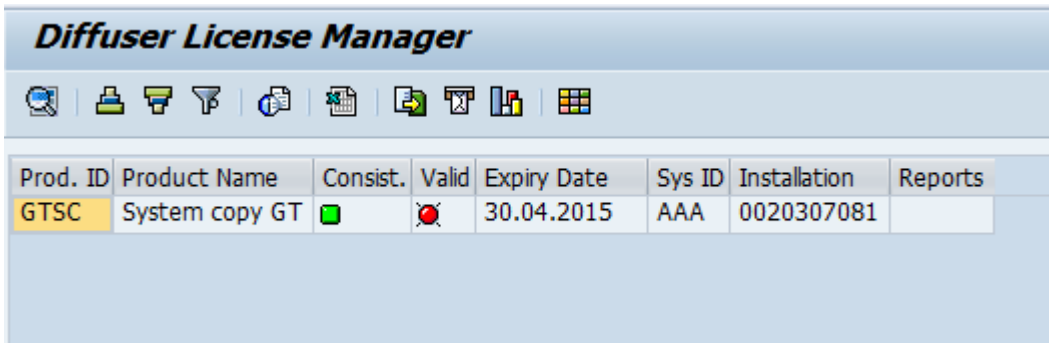
After entering the license screens through the transaction /N/BTR/LICENSE or via the button on the /N/BTR/MINICUBE as described in Maintaining Licenses.

To install a license simply select the “Install key file” option as below and execute.

Select the file you want to install. The filename will contain a 10 digit number for the SAP installation number of the intended system.
There may also be the three character system ID and the product name.

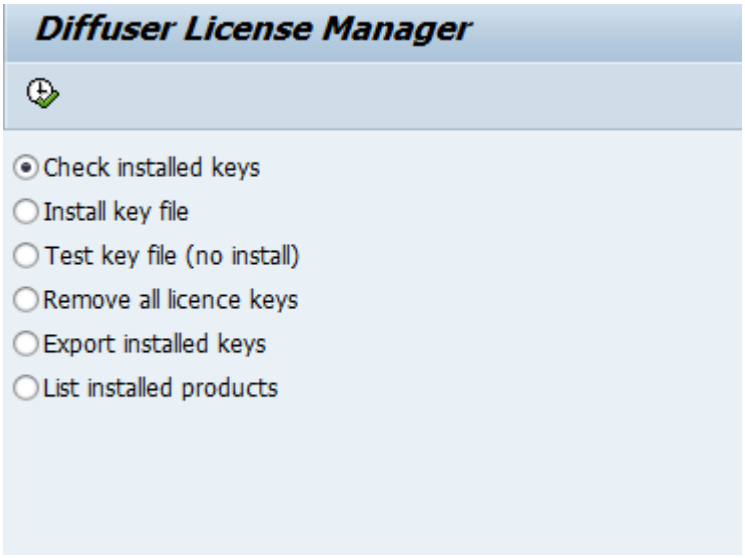


If errors are found then the problem is displayed.

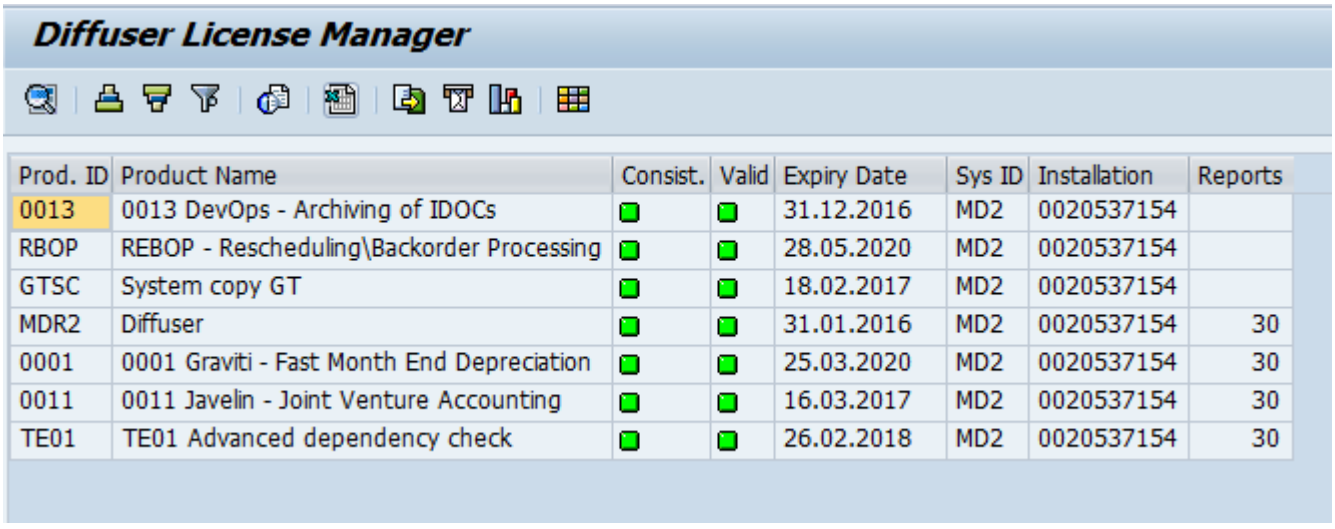


Check installed keys

To check the products you have installed in your system, select the “List installed products” option as below and Execute.



The keys installed are shown as below. Note that any red status icons indicate a problem with the key details being consistent with the system or being out of validity.



Prod. ID	Product Name	Consist.	Valid	Expiry Date	Sys ID	Installation	Reports
0013	0013 DevOps - Archiving of IDOCs			31.12.2016	MD2	0020537154	
RBOP	REBOP - Rescheduling\Backorder Processing			28.05.2020	MD2	0020537154	
GTSC	System copy GT			18.02.2017	MD2	0020537154	
MDR2	Diffuser			31.01.2016	MD2	0020537154	30
0001	0001 Graviti - Fast Month End Depreciation			25.03.2020	MD2	0020537154	30
0011	0011 Javelin - Joint Venture Accounting			16.03.2017	MD2	0020537154	30
TE01	TE01 Advanced dependency check			26.02.2018	MD2	0020537154	30

List installed products

To check the products you have installed in your system, select the “List installed products” option as below and execute.

A list of installed products on your system is shown.

Diffuser License Manager

		ObjectTypeName
0001	0001 Graviti - Fast Month End Depreciation	/BTR/CL_MDR_LICENCE_KEY_PP_DEP
0002	0002 Consenti - Compliance Control Engine	/BTR/CL_MDR_LICENCE_KEY_PP_CGT
0003	0003 ExPo - Fast PO Status Tracker	/BTR/CL_MDR_LICENCE_KEY_PP_EXP
0004	0004 PoGo - Fast PO Closure	/BTR/CL_MDR_LICENCE_KEY_PP_POC
0006	0006 Setelite - Fast Month End Settlement	/BTR/CL_MDR_LICENCE_KEY_PP_SET
0007	0007 Articlus - Fast Retail Assortment Publisher	/BTR/CL_MDR_LICENCE_KEY_PP_ART
0008	0008 Production Order Settlement	/BTR/CL_MDR_LICENCE_KEY_PP_STP
0009	0009 SnapOps - Scramble	/BTR/CL_MDR_LICENCE_KEY_PP_DSF
0010	0010 BDEX	/BTI/CL_MDR_LICENCE_KEY_PP_BDX
0011	0011 Javelin - Joint Venture Accounting	/BTR/CL_MDR_LICENCE_KEY_PP_JVA
0012	0012 Batch Accelerator	/BTR/CL_MDR_LICENCE_KEY_BATCH
0013	0013 DevOps - Archiving of IDOCs	/BTR/CL_MDR_LICENCE_KEY_PP_ARI
0014	0014 Fast Material Document List	/BTR/CL_MDR_LIC_KEY_MAT_LIST
0015	0015 DevOps - Archiving of Sales Orders	/BTR/CL_MDR_LIC_KEY_PP_VBAK
0016	0016 DevOps - Archiving of FI Docs	/BTR/CL_MDR_LIC_KEY_PP_FIDOC
0017	0017 DevOps - Emma Case	/BTR/CL_MDR_LIC_KEY_PP_EMMACAS
0018	0018 DevOps - Archiving of Billing Doc	/BTR/CL_MDR_LIC_KEY_PP_VBRK
0019	0019 DevOps - Archive Delete	/BTR/CL_MDR_LIC_KEY_PP_ARCHDEL
0020	0020 DevOps -Archiving of EMMA Job	/BTR/CL_MDR_LIC_KEY_PP_EMMAJOB
DRCC	Remote Client Copy GT	/BTR/CL_MDR_LICENCE_KEY_FRCC
GT	Description	/BTR/CL_MDR_LICENCE_KEY_GT0003
GTSC	System copy GT	/BTR/CL_MDR_LICENCE_KEY_GT_SC
MDR2	Diffuser	
RBOP	REBOP - Rescheduling\Backorder Processing	/BTR/CL_MDR_LICENCE_KEY_PP_BOP
TE01	TE01 Advanced dependency check	/BTI/CL_MDR_LICENCE_KEY_TE_ADC

When to use Diffuser

Node5 Diffuser is essential when the data volumes are so large that the processing time to run your programs is unacceptable. If you are able to write a report that runs within acceptable time-constraints then the use of Diffuser may not be required. However, even if your report runs within 2 hours (and this is considered acceptable) you are still able to use Diffuser to bring this run-time down even further.

It is our consideration that almost any program that executes in the background can gain benefits by being implemented in Node5 Diffuser. This is because the Node5 Diffuser format not only promotes performance improvements, but also cost reductions in maintenance by having a generic format.
It provides the benefit of separating the processing and presentation logic. Separate presentation logic allows the data to be viewed interactively through Node5 MiniCube and in a user-friendly manner, far easier than lengthy static list or spool output.

Setting up a Diffuser program

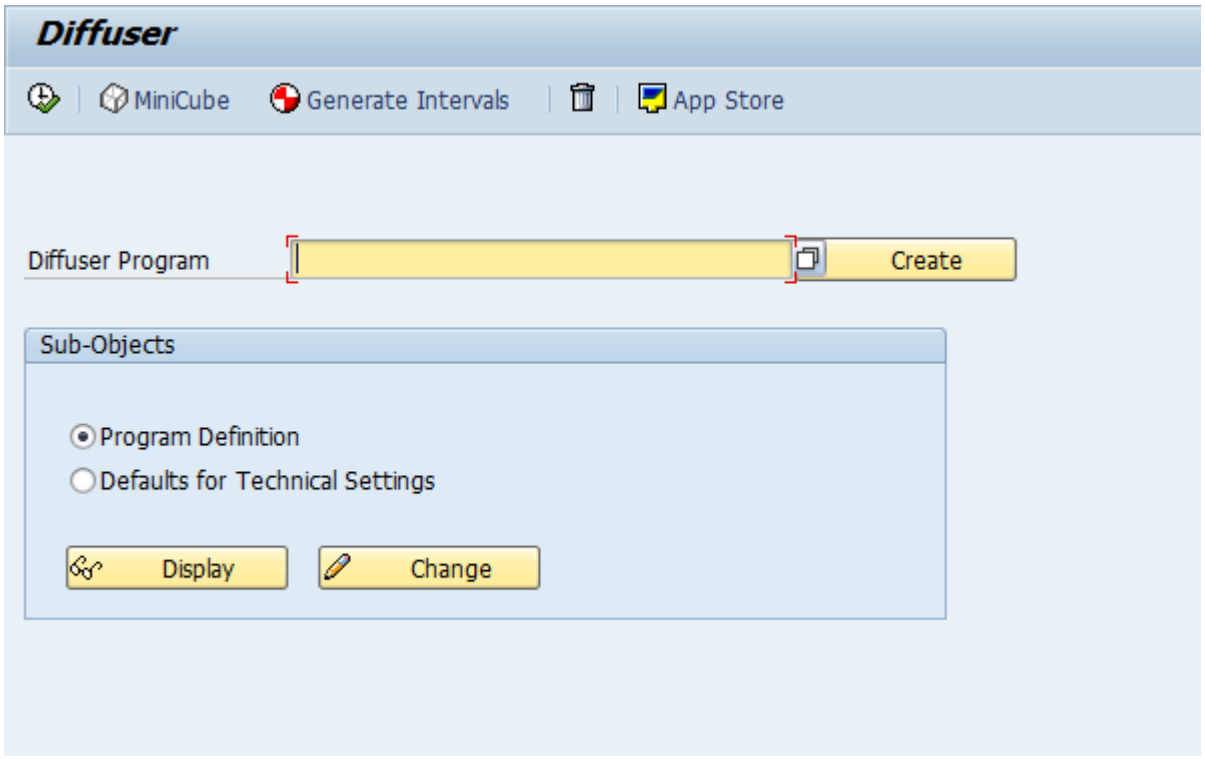
For a program to use the Node5 Diffuser, it can either be developed as a custom Z Accelerator or provided as a prepackaged program supplied by Basis Technologies (as a GT, GTi or BDi App).

To setup a program to use Node5 Diffuser use the transaction /BTR/DIFFUSER here as a minimum define the Diffuser program in the Program Definition, then choose to setup the technical settings and generate intervals where required.

- Program Definition
- Default Technical Settings
- Interval Generation

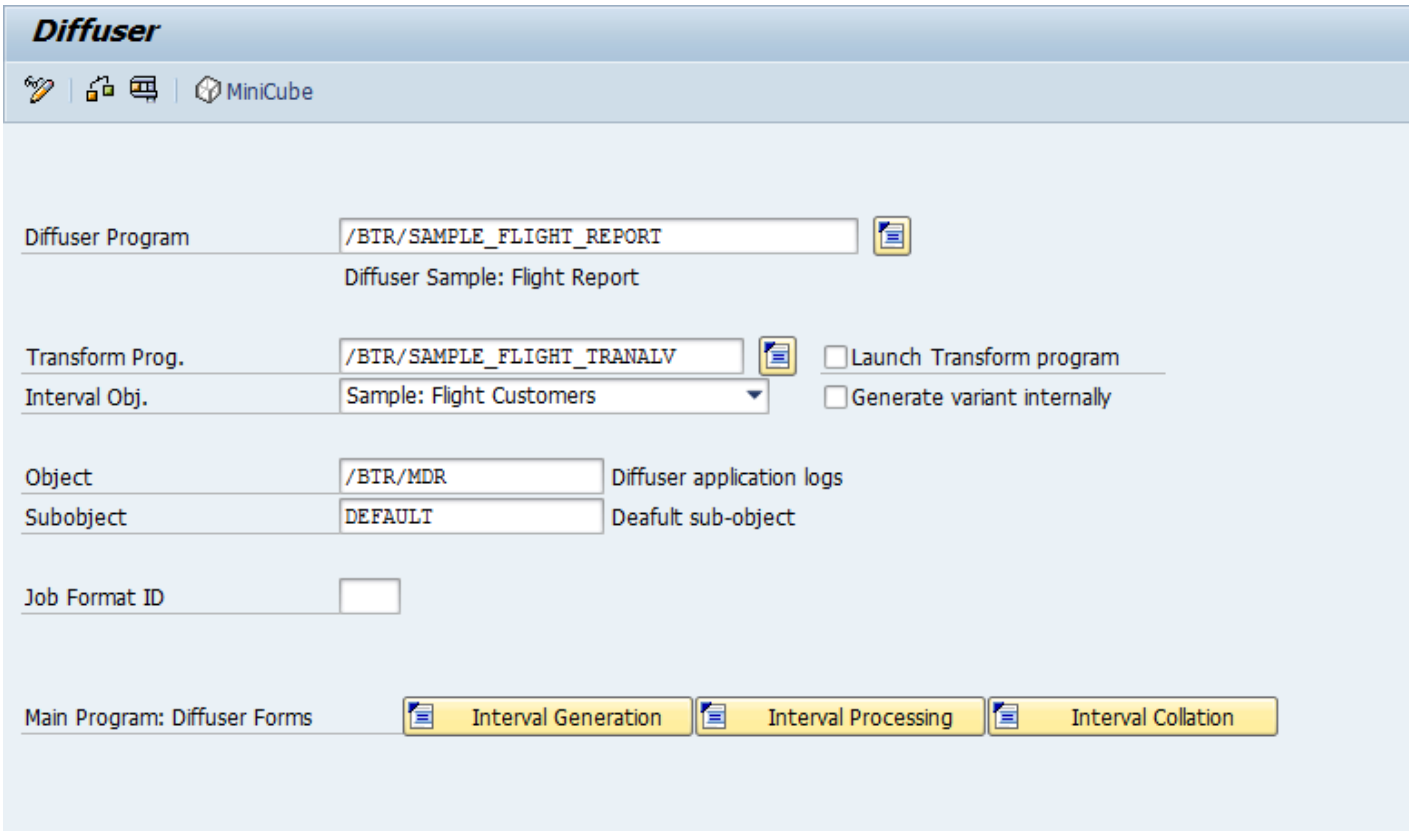
Program Definition

The definition of an Node5 Diffuser program is set up via the transaction /n/BTR/DIFFUSER.



Enter the program name and press the Create button for new programs or Change button for an existing program with the sub-object as program definition.

The definition is now displayed as below.



The main program is already populated from the first screen. A transformation program can now be configured if required.

The “Interval Object” object is also populated here, refer here for more information on intervals.

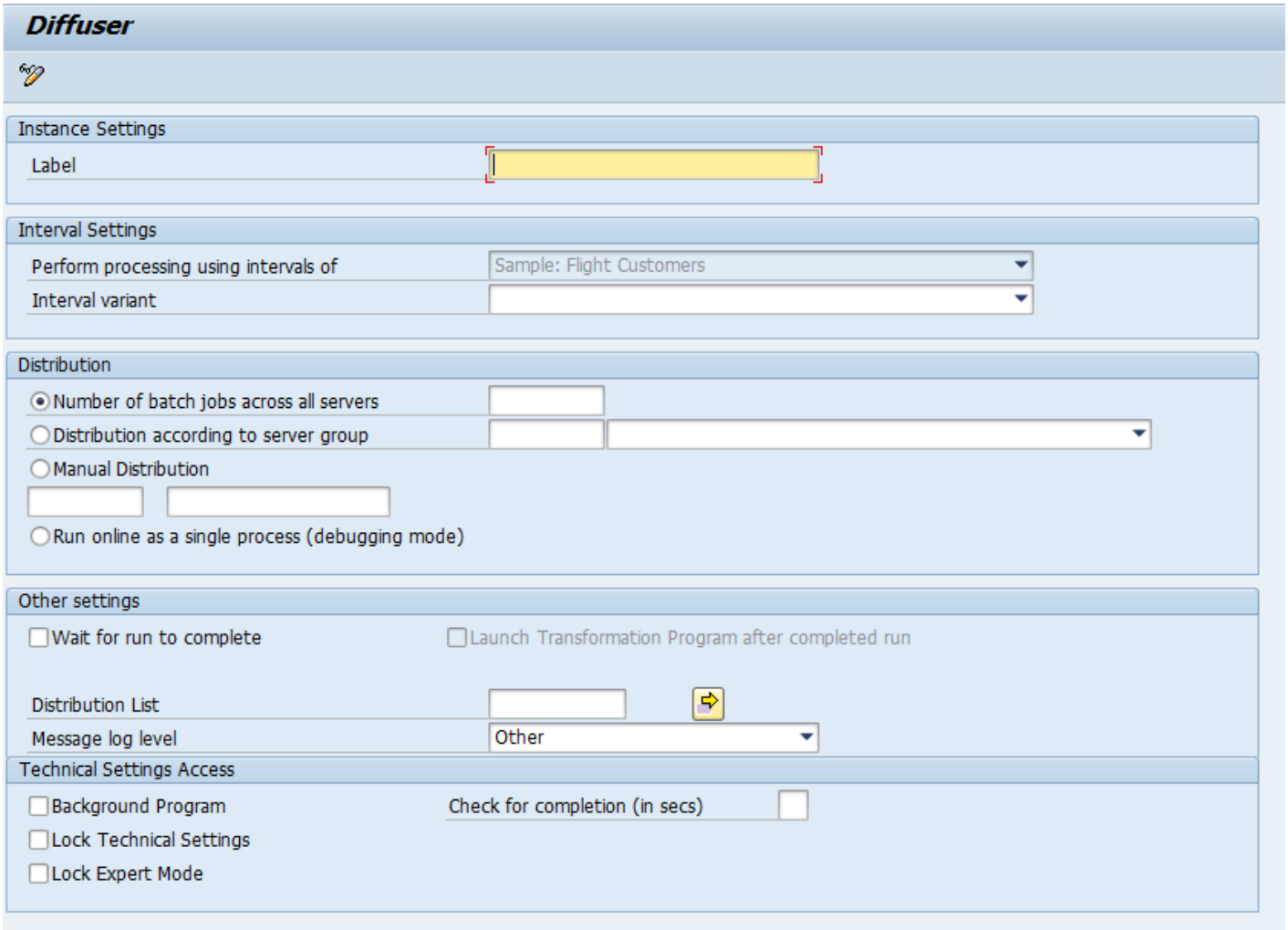
The program definition also allows the user to configure which application log object and sub-object any messages are written to that are called during the execution of the program. The default object and sub-object are /BTR/MDR/ and DEFAULT respectively.

Furthermore, the transaction N/BTR/DIFFUSER allows the developer to maintain the Main Program and Transformation Program directly instead of using the standard SAP transaction SE38, with the code buttons at the bottom linking directly to the relevant subroutines.

Default Technical Settings

The second sub-object managed through the transaction /BTR/DIFFUSER is “Defaults for Technical Settings”. This screen contains two main sections.

1. “Defaults for Technical Settings” allows to set default values for a specific Diffuser program. Once set, these values will always appear on the Technical Settings’ sub screen for that program see Technical Settings under Running Diffuser Programs for more details.



1. “Technical Settings Access” we will explore in more detail.

Background Program

NOTE: This functionality requires “Wait for run to complete” to be set.

This option supresses the default display of the Run History screen after completion of an instance run. This is a useful feature that allows a Diffuser program to be called from another program without interrupting the latter with the MiniCube display.

The input field “Check for completion (in sec)” allows to set in seconds a time interval in which the parent job of a running Diffuser instance will wake up and check if all child parallel processes have completed. The default wake up and check time is 30 seconds which is suitable for very long running programs but not for speeding up web services where every second counts.

Lock Technical Settings

This options allows to lock all input fields for Technical Settings. This is useful if when a program can repeatedly run with the same default values and users should not change those values. When this option is set, the Diffuser Mode in the MiniCube will be locked as well.

This restriction applies at program level and not at user level. That is, once set the Technical Settings will be locked for all users. Restrictions at user level can be implemented, see the section Authority Checks in the Z Accelerators Guide.

Lock Expert Mode

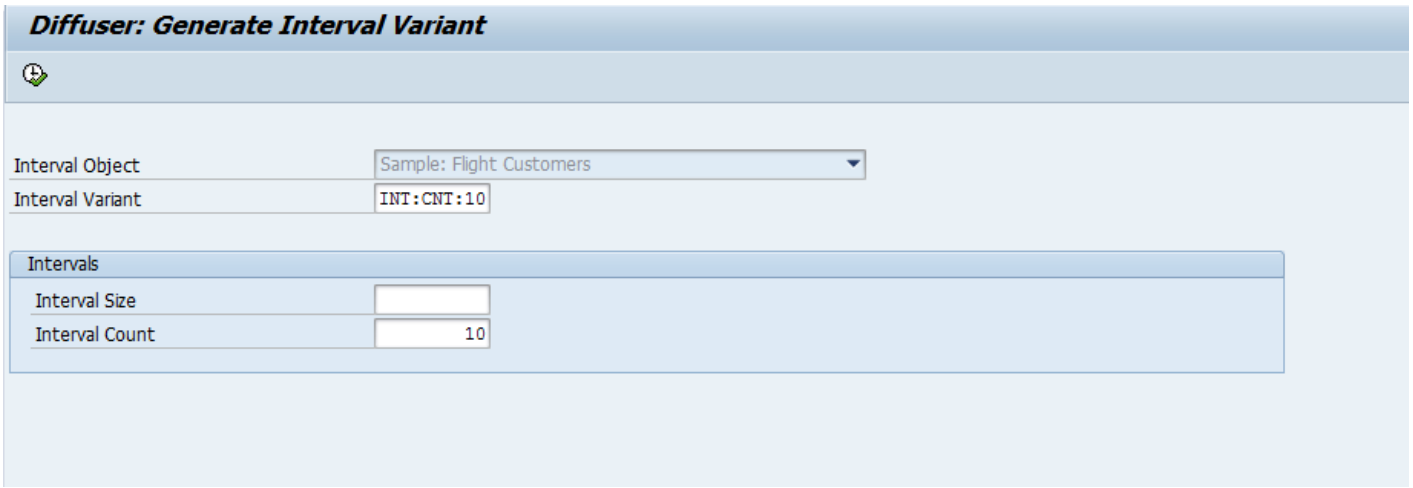
This option is similar to “Lock Technical Settings”. The only difference is that on the Technical settings screen only the input fields under “Distribution” are locked. This allows the user to change settings like label name while protecting the more critical job distribution section from potential misuse. This option applies at program level as well. Restrictions at user level can be implemented with see the section Authority Checks in the Z Accelerators Guide.

Interval Generation

When a Node5 Diffuser program uses Interval Objects, an Interval Variant needs to be created from the Interval Object, before the Diffuser program is run. An Interval Variant can be thought of as the set of Intervals that the Diffuser program is going to use. It is necessary that new Interval Variants are generated regularly (potentially before each batch run) to ensure that the intervals are split evenly as the data in the system grows.

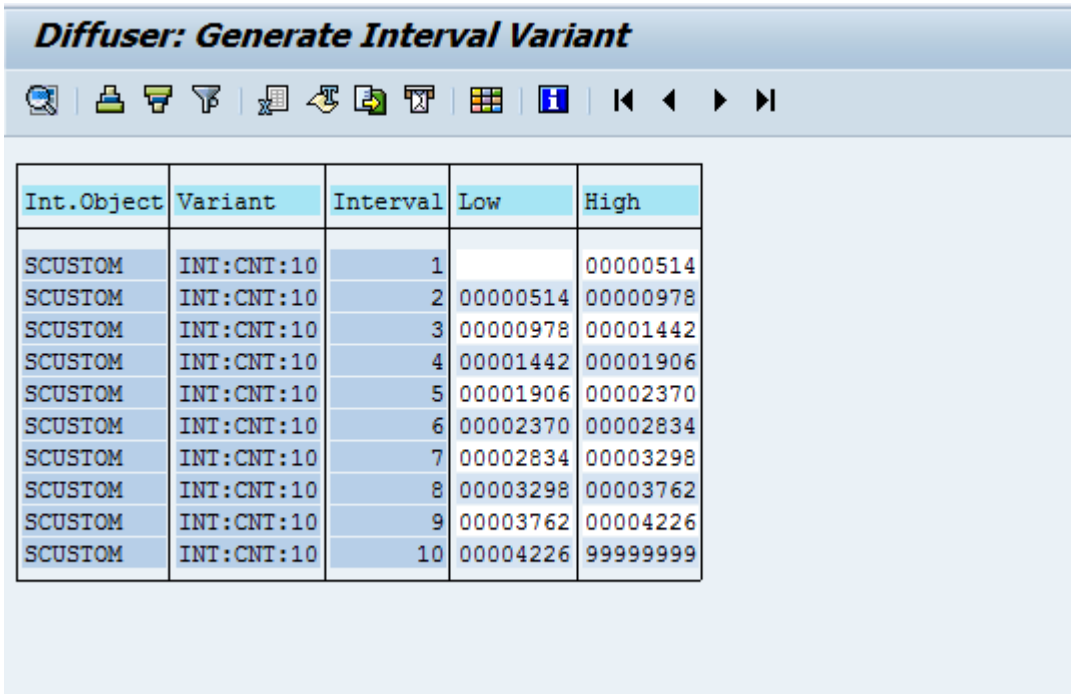
To use an Interval Objects, they must first be configured into the framework via table /BTR/INTVALOBJ.

There are two different types of Interval Objects; standard SAP Mass Run Interval Objects and Diffuser Interval Objects. They have similar operation except for the generation of the Interval Variants. The Intervals (or Interval Variant) are created before the Diffuser program is executed; this is done via the program /BTR/MDR_INTERVAL_GENERATION or alternately for standard SAP Interval variants via transaction FQD2. Either an Interval Count or Interval Size can be used as parameters to how the Intervals get generated.



The above example shows the generation of a new Interval Variant called INT:CNT:10 from the Interval Object “Customer Interval Object”, with the requirement that 10 Intervals are to be created.

As can be seen below, the result is 10 generated Intervals.



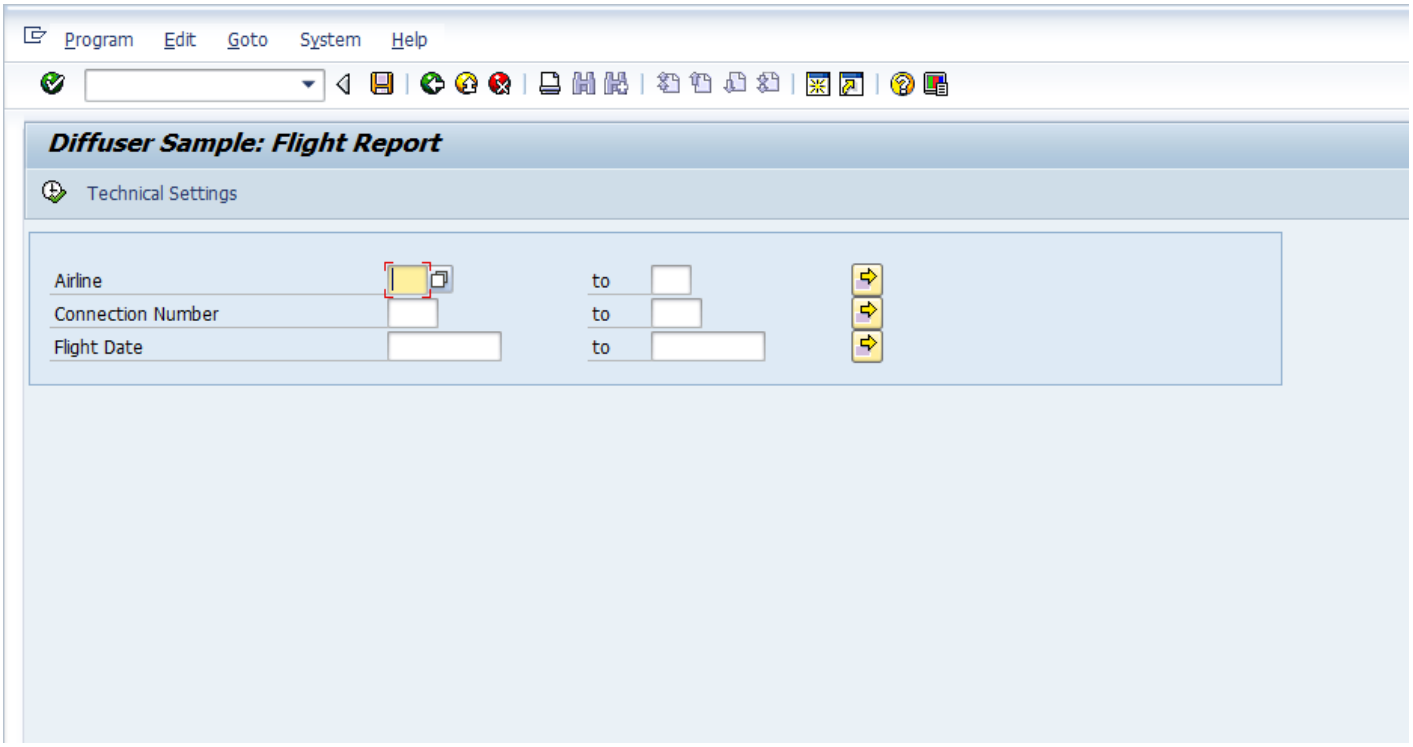
Diffuser: Generate Interval Variant

Int.Object	Variant	Interval	Low	High
SCUSTOM	INT:CNT:10	1		00000514
SCUSTOM	INT:CNT:10	2	00000514	00000978
SCUSTOM	INT:CNT:10	3	00000978	00001442
SCUSTOM	INT:CNT:10	4	00001442	00001906
SCUSTOM	INT:CNT:10	5	00001906	00002370
SCUSTOM	INT:CNT:10	6	00002370	00002834
SCUSTOM	INT:CNT:10	7	00002834	00003298
SCUSTOM	INT:CNT:10	8	00003298	00003762
SCUSTOM	INT:CNT:10	9	00003762	00004226
SCUSTOM	INT:CNT:10	10	00004226	99999999

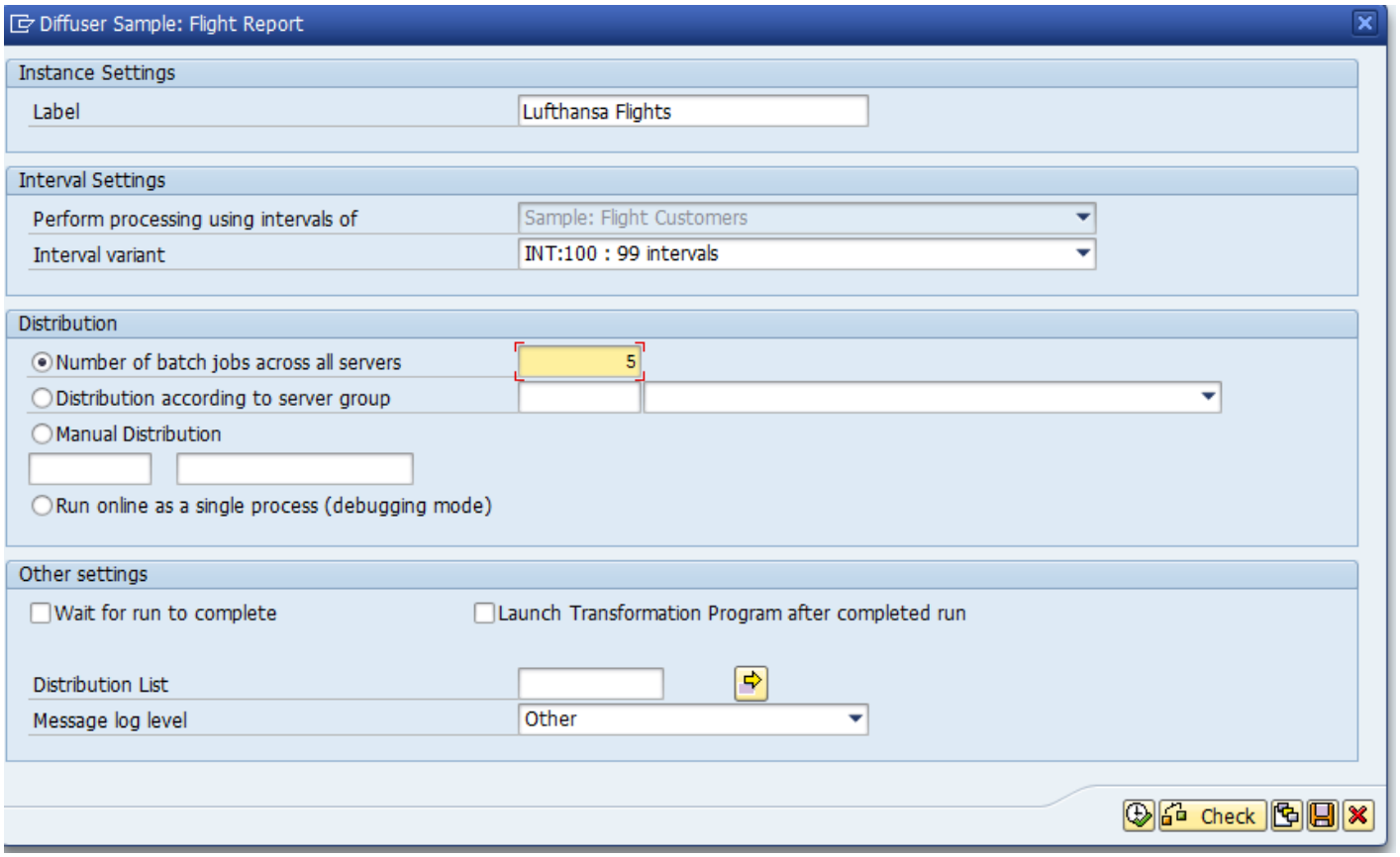
Running Diffuser Programs

Technical Settings

The key part that the user sees is the “Technical Settings” button as below.



If you select the “Technical Settings” button, you will be prompted for the Node5 Diffuser specific technical settings.



These “Technical Settings” are important when executing an Diffuser program be it in a production environment, or when performing unit testing of your Diffuser program.

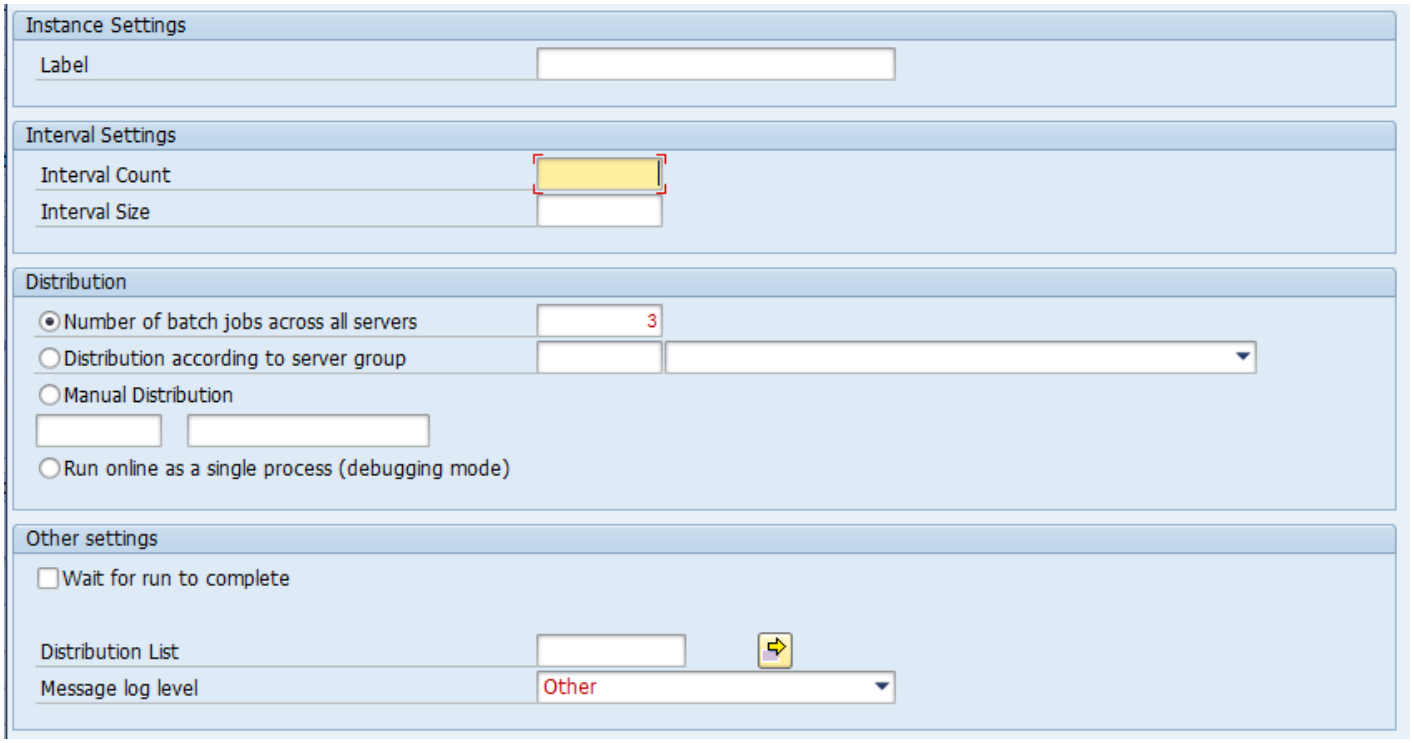
Note that you can set up Default Technical Settings and set up user authorizations to control the users ability to change these settings, for more information refer to the Z Accelerators – Developers Guide Authority Checks.

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

- Label – The first is a label that can be specified to identify this particular execution.
- Perform processing using intervals of – This is the interval object confirmed in the Program Definition
- Interval Variant – The Interval variant provides you with a list of different pre-generated Intervals. As detailed in the section Interval Generation, the interval variants are pre-generated using program /BTR/MDR_INTERVAL_GENERATION
- 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, in a foreground process.
- 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.
- Launch Transformation Program after completed run – This means the MiniCube screen is skipped so the user gets straight to their results when the run completes
- 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.

When using Dynamic Intervals as set out in the Z Accelerators – Developers Guide Dynamic Interval Generation the “Interval Settings” section will change and be replaced with the two options Interval Count and Interval Size introduced, this looks as per the screenshot below.

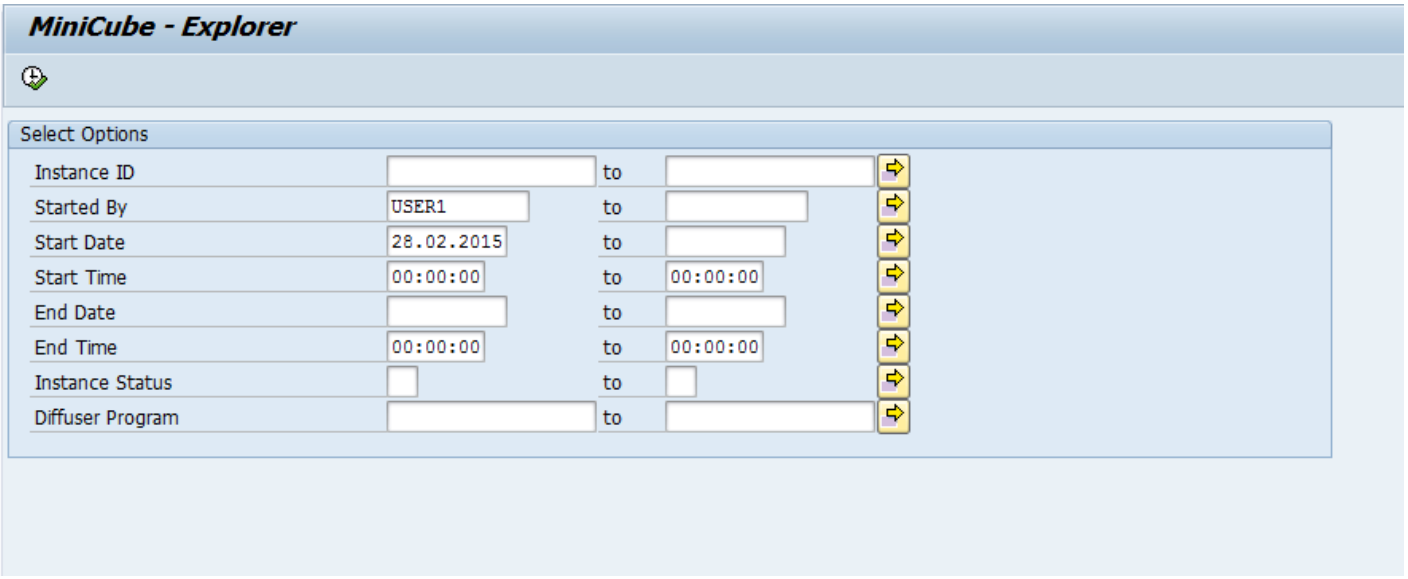


The impact of the two fields is as below:

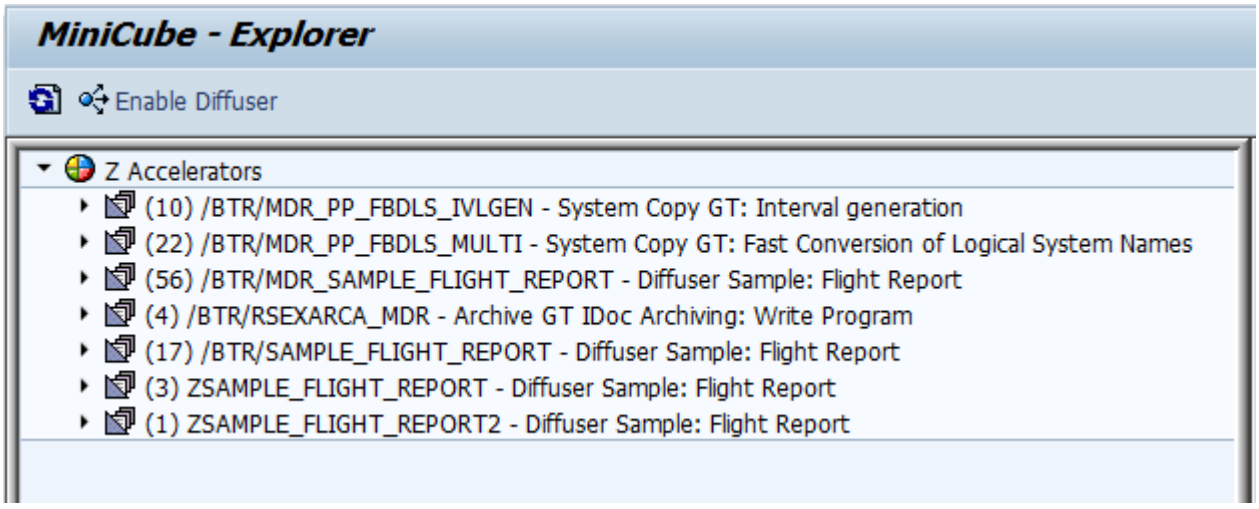
- Interval Count – This specifies the number of intervals (pieces) that the total amount of work to be done is to be broken up into
- Interval Size – This specifies the “number of objects” to be put into each interval to be then worked upon independently

Accessing MiniCubes

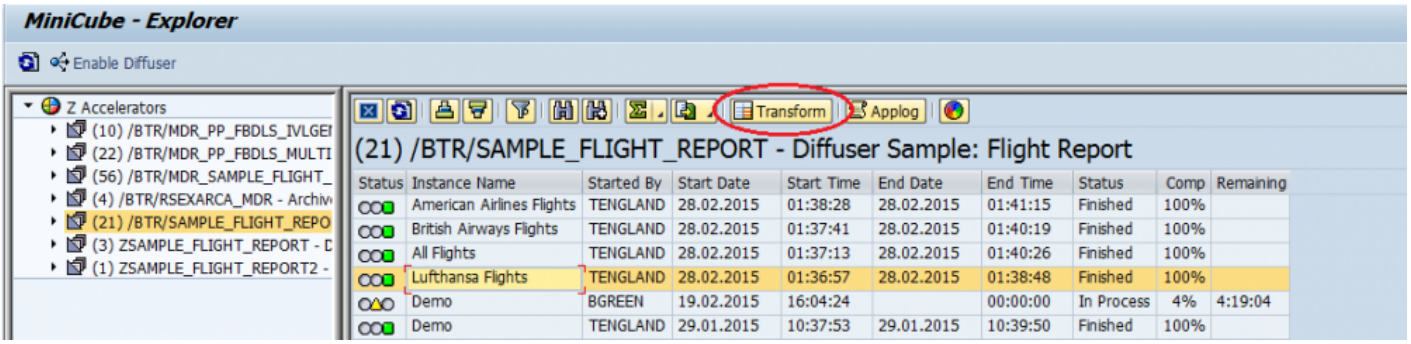
Historical instance runs and results can be easily accessed via transaction /BTR/MINICUBE. It allows to search by user, time period, status and program. This is especially useful for making result data available to users without having to rerun the programs. In the selection screen insert your search criteria and execute.



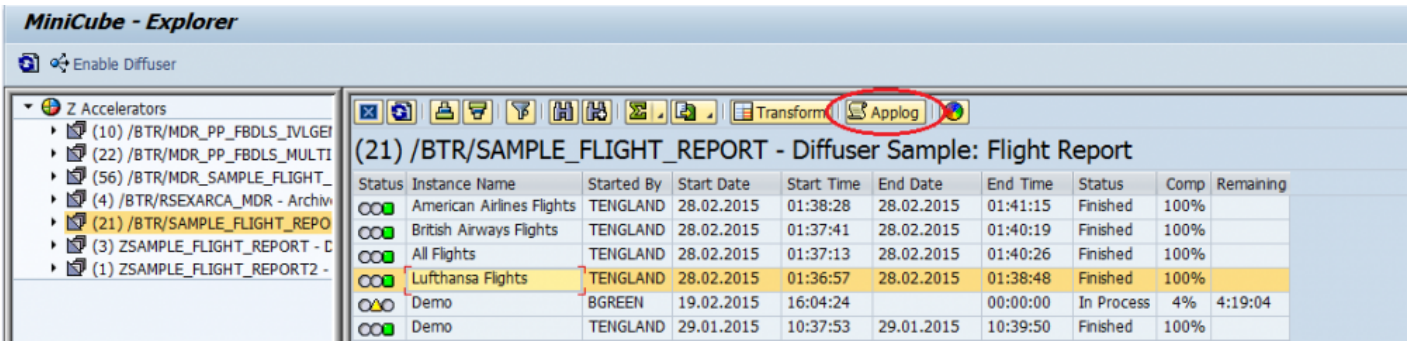
MiniCube will show a list of the Diffuser defined program(s) with instances relevant to the search criteria, expand the Z Accelerators Node to reveal the results.



By drilling down on the program name the user will access the programs instance runs. Select an instance and double click or click “Transform” to display the results of the run.



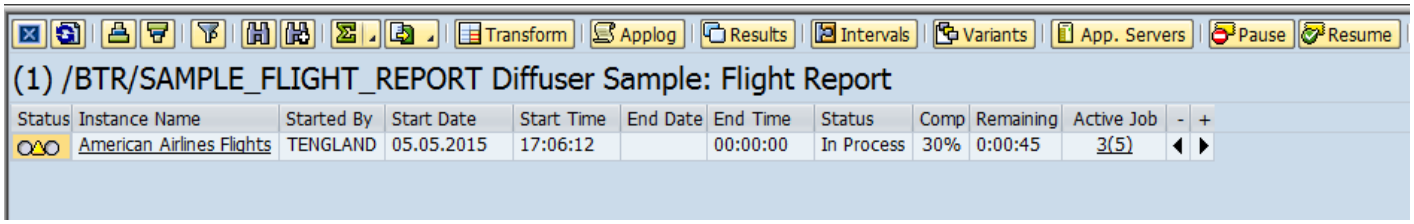
In the same manner you can check the application log for error messages.



Once on the screen above the user will be able to see and administer historical data as well as instances in progress using the functionality mentioned in Administering Diffuser Programs.

Administering Diffuser Programs

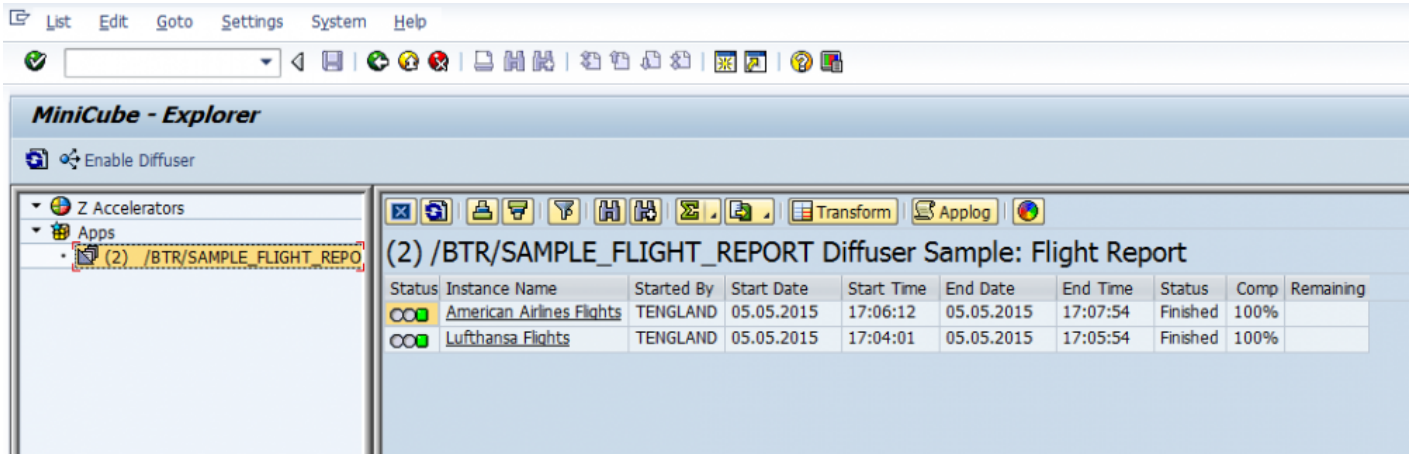
The Node5 Architecture provides the advanced user a number of powerful administrative capabilities via the MiniCube transaction /N/BTR/MINICUBE (see screen below). These capabilities provide a powerful way of managing your Diffuser programs.



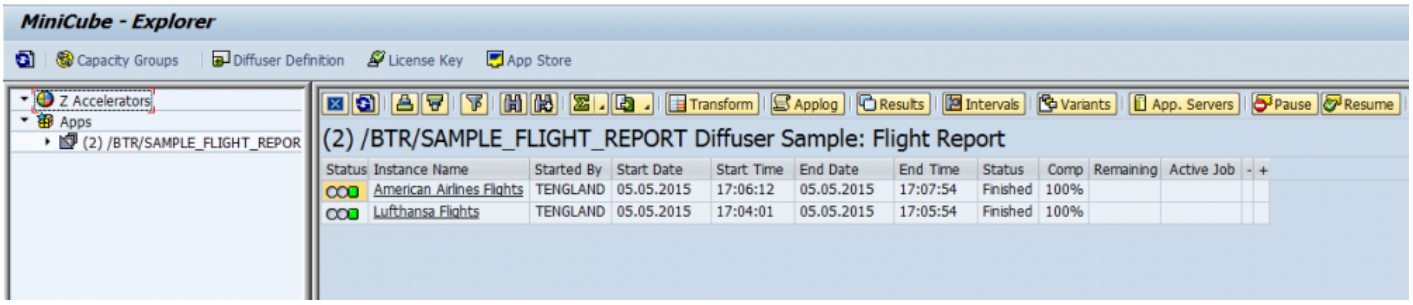
- Diffuser Mode
- Intervals
- Results
- Variants
- App Servers
- Increase or Decrease Jobs
- Pause
- Resume
- Delete
- Force Error
- Reprocess Error
- Debug an Interval
- Rename Instance

Diffuser Mode

To access the expert mode click the Diffuser mode button as below.

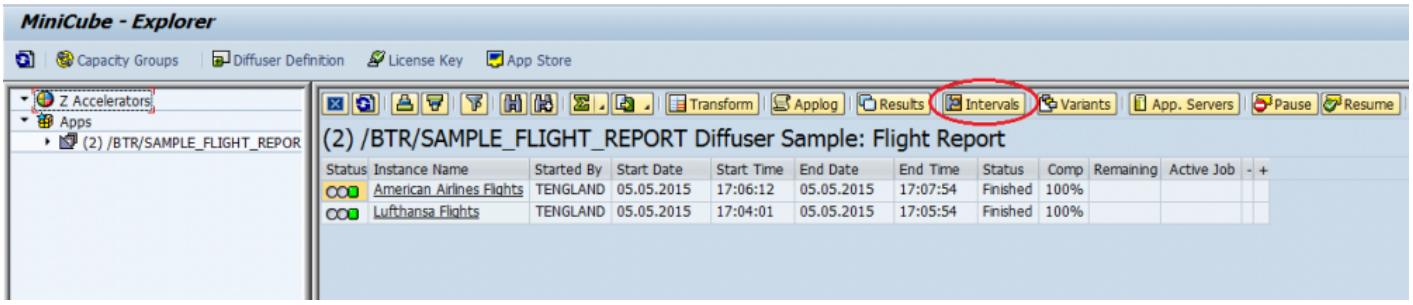


If authorized a number of other functions will be revealed.

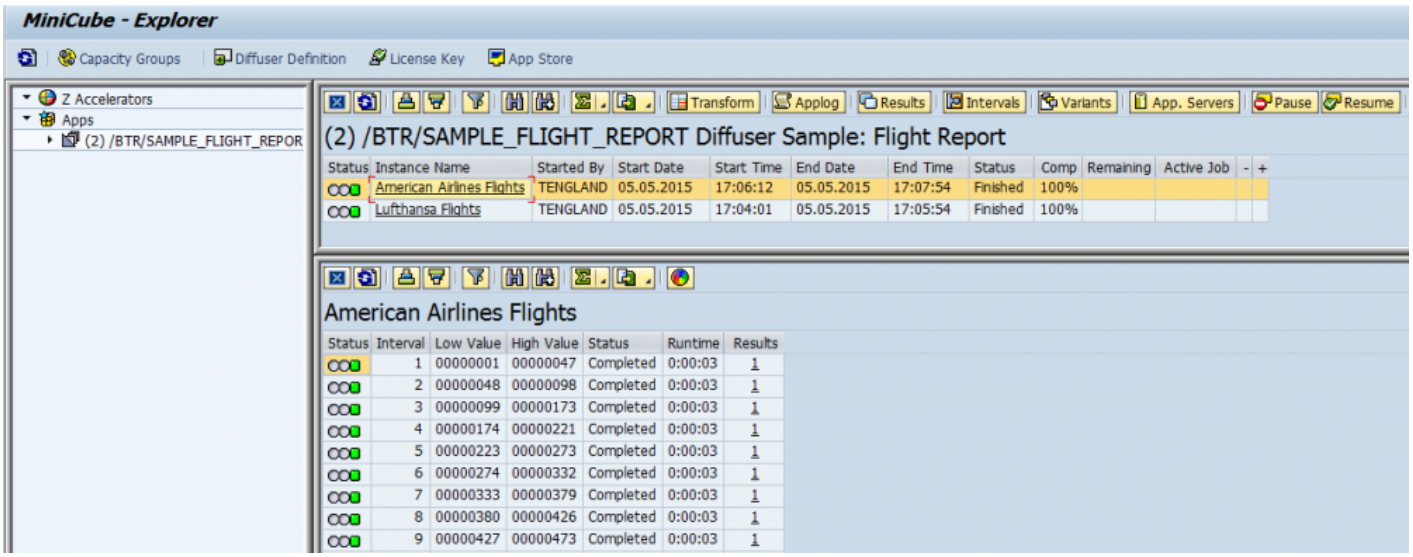


Intervals

By drilling down on the program name the user will access the programs instance runs. Select an instance and in Diffuser mode double click the instance or click “Intervals” to display the intervals to that specific instance run.

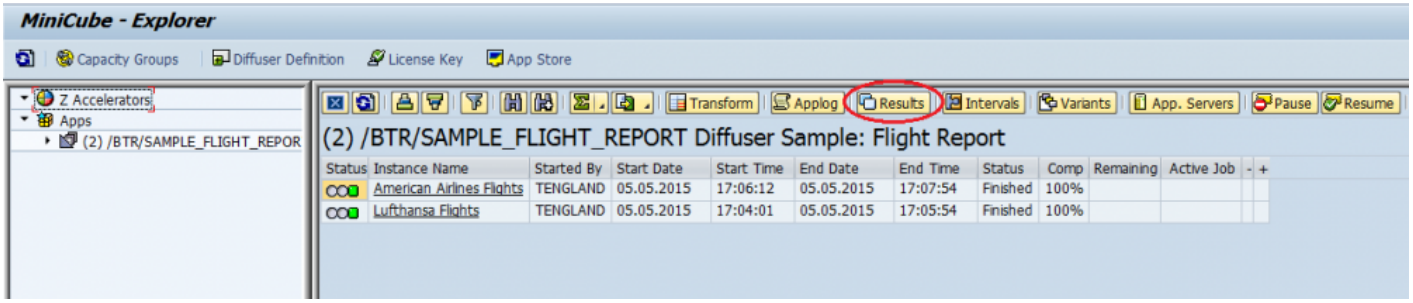


The details of all the intervals are then displayed as below.

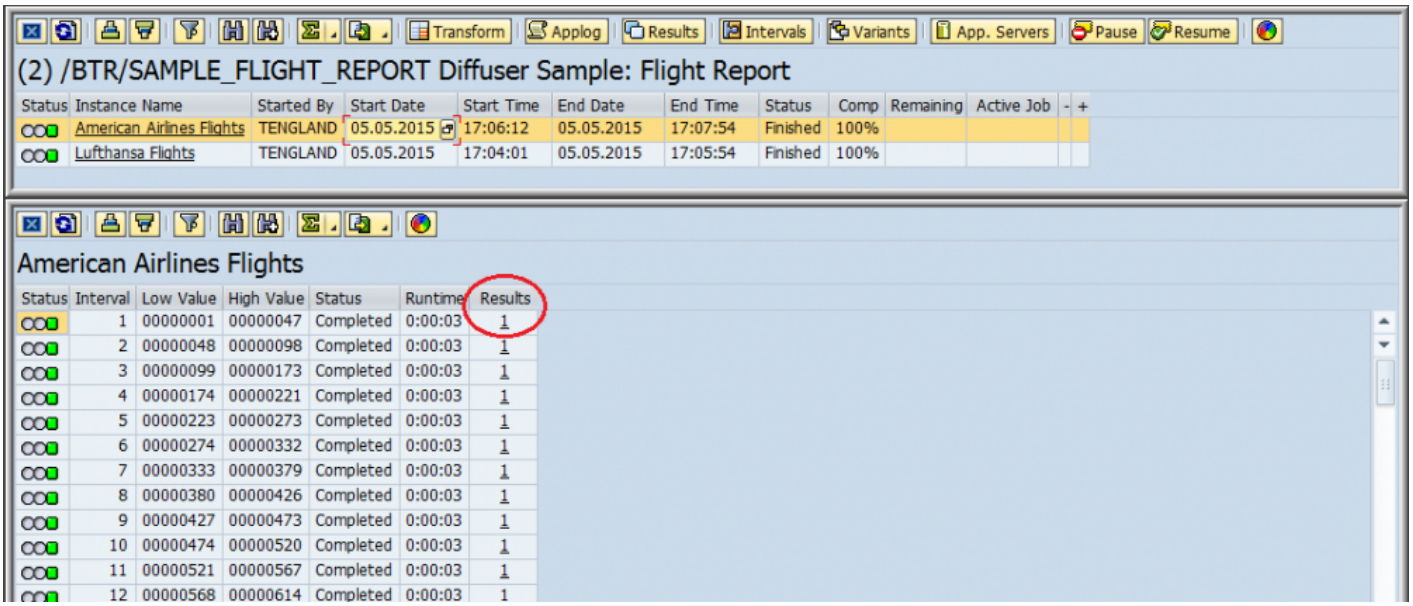


Results

To access the raw results stored against the instance click the results button as below.

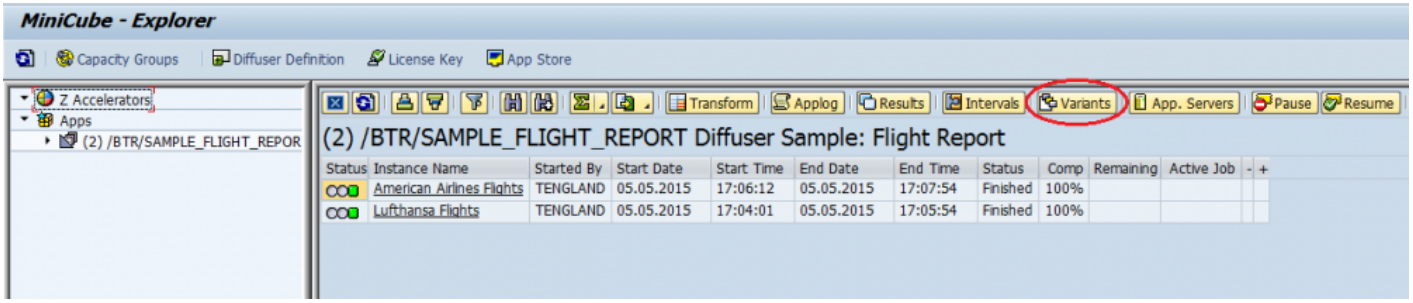


You can also select an interval and view the raw results stored against each interval, by double-clicking the number of results.

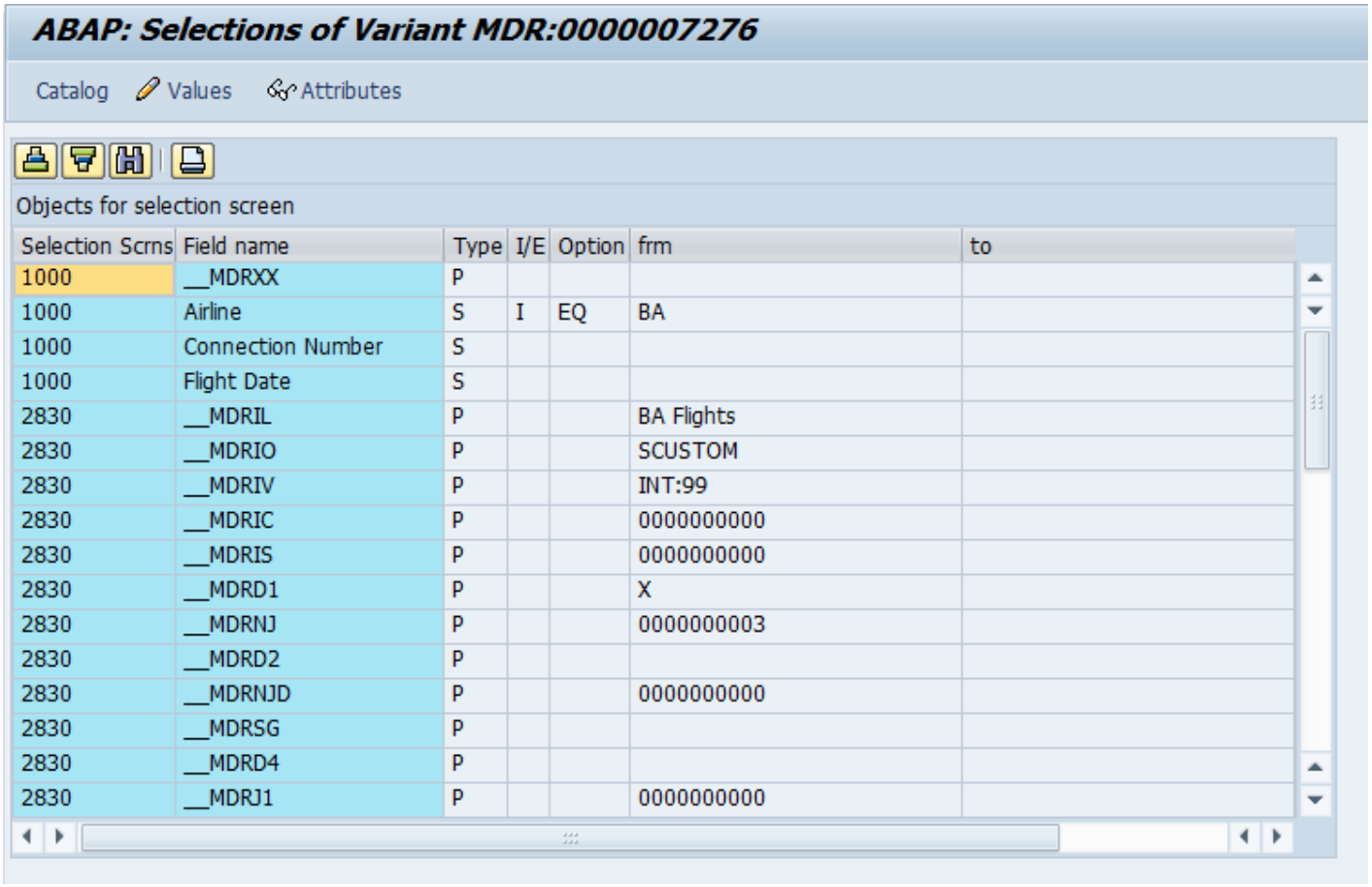


Variants

To access the details entered on the selection screen for an instance click the variant button as below.



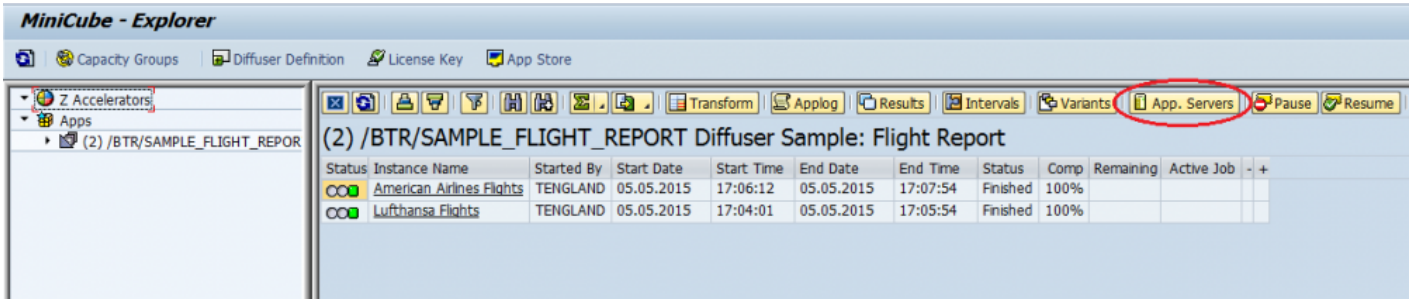
This enables the variant details entered on the selection screen to be viewed.



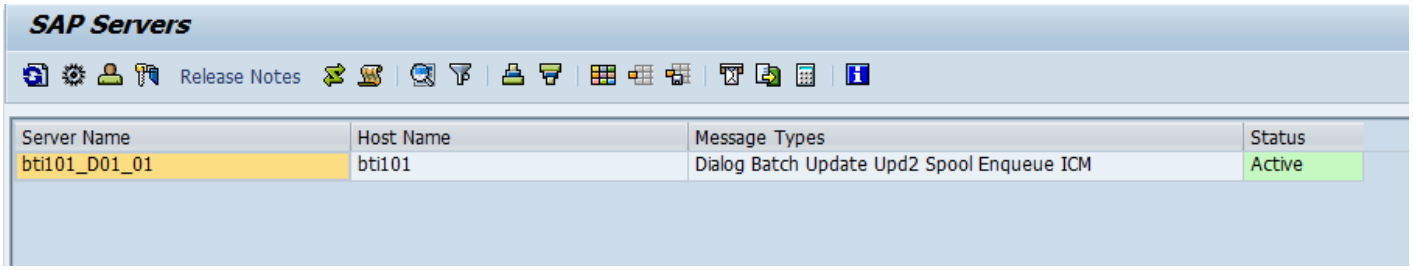
Selection Scrns	Field name	Type	I/E	Option	frm	to
1000	__MDRXX	P				
1000	Airline	S	I	EQ	BA	
1000	Connection Number	S				
1000	Flight Date	S				
2830	__MDRIL	P			BA Flights	
2830	__MDRIO	P			SCUSTOM	
2830	__MDRIV	P			INT:99	
2830	__MDRIC	P			0000000000	
2830	__MDRIS	P			0000000000	
2830	__MDRD1	P			X	
2830	__MDRNJ	P			0000000003	
2830	__MDRD2	P				
2830	__MDRNJD	P			0000000000	
2830	__MDRSG	P				
2830	__MDRD4	P				
2830	__MDRJ1	P			0000000000	

App Servers

To view the application servers click the App Server button as below.



This then displays the available App Servers

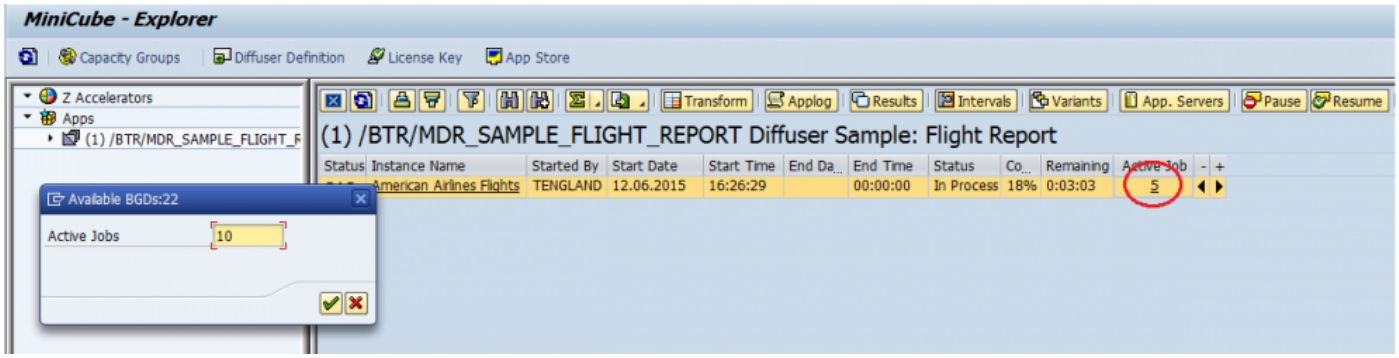


Increase or Decrease Jobs

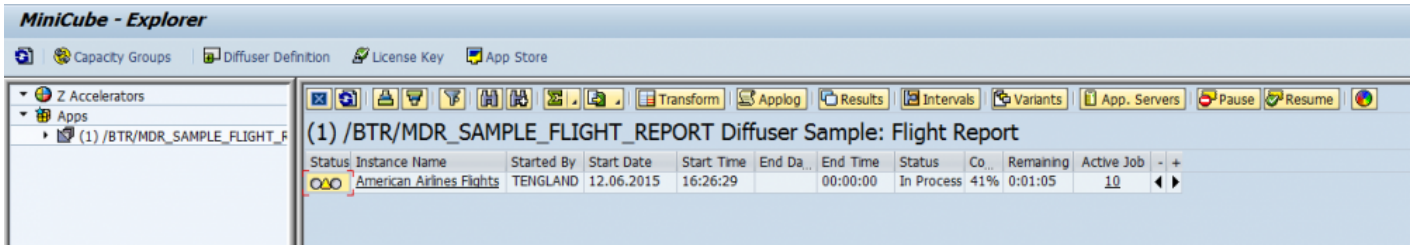
Through the MiniCube transaction, you can see historical instances of a program as well as any currently executing program instances. You can also see the number of active jobs for each program instance currently running, and it is possible to change the number of jobs running for a particular active job.

Adding more jobs can help decrease the run time.

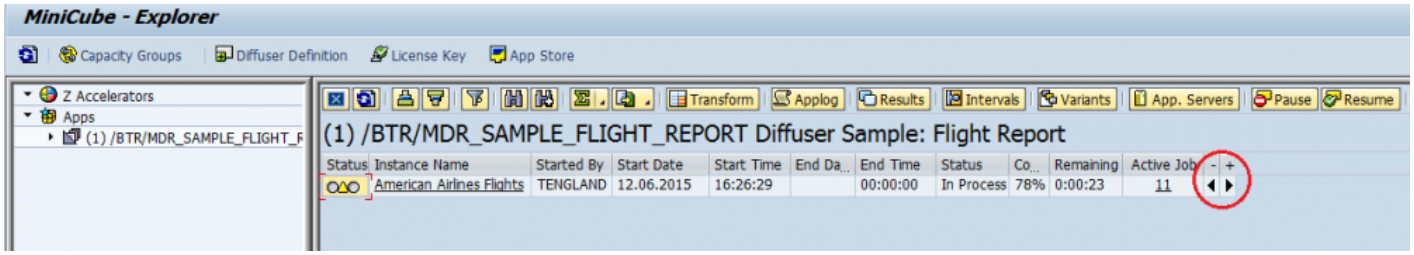
To change the number of jobs click the number of jobs currently running and a popup appears where you can enter the new number of jobs you want the instance to run. Note the top of the popup box shows the number of unused background jobs in the system at that point in time, in this case 22.



Instance is now running 10 jobs.



Alternatively you can click the arrow buttons to increase or decrease the jobs one at a time.

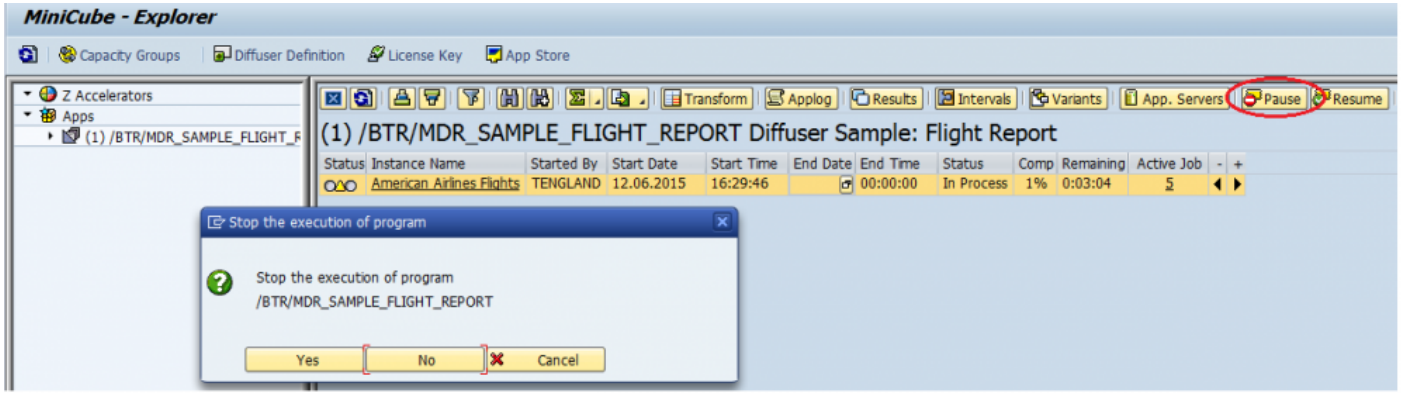


Pause

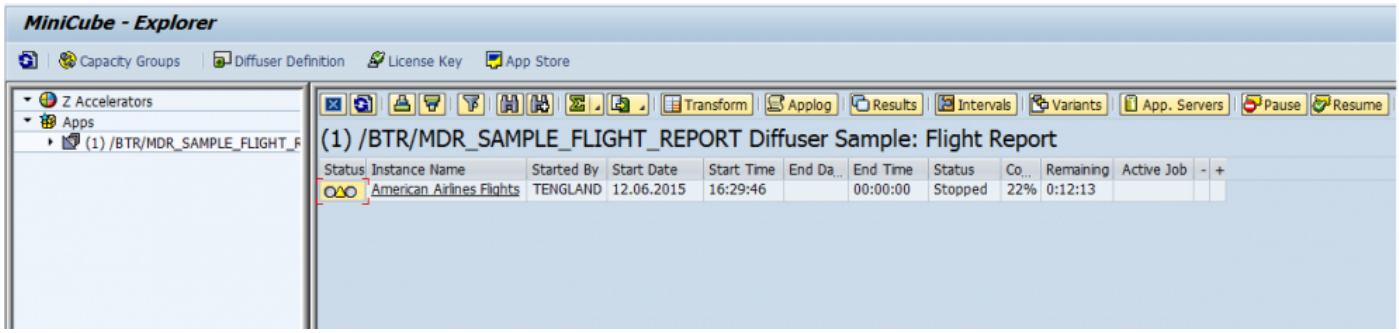
It is possible to Pause (or Stop) a program instance using this option. By selecting this option after selecting a program instance, the Diffuser tells the currently executing jobs to no-longer process any more intervals after it completes the processing of the current intervals. The status of the Instance, and unprocessed intervals changes to “Stopped”. You will need to click Refresh to update the status. This is a powerful option that is used typically when a Diffuser program needs to be stopped temporarily due to the need to free up batch resources, or stopped permanently if the report run is no longer required. When the instance is paused, the Diffuser framework will not immediately stop all jobs that are currently running. It will instead prevent any new intervals from being started. The more intervals there are the more control over the execution of the instance an administrator will have.

The benefit Diffuser has over the traditional approach to executing reports is that the Diffuser program does not need to start over again, execution can continue from where it left off. The intervals that have already been processed do not need to be reprocessed unless of course it is deemed necessary by the user due to perhaps a substantial amount of time passing before the program is allowed to continue. It is only possible to pause an Instance that is currently in the “In Progress” status.

To pause a program simply select the instance and hit the pause button, you will be asked to confirm that you want to.



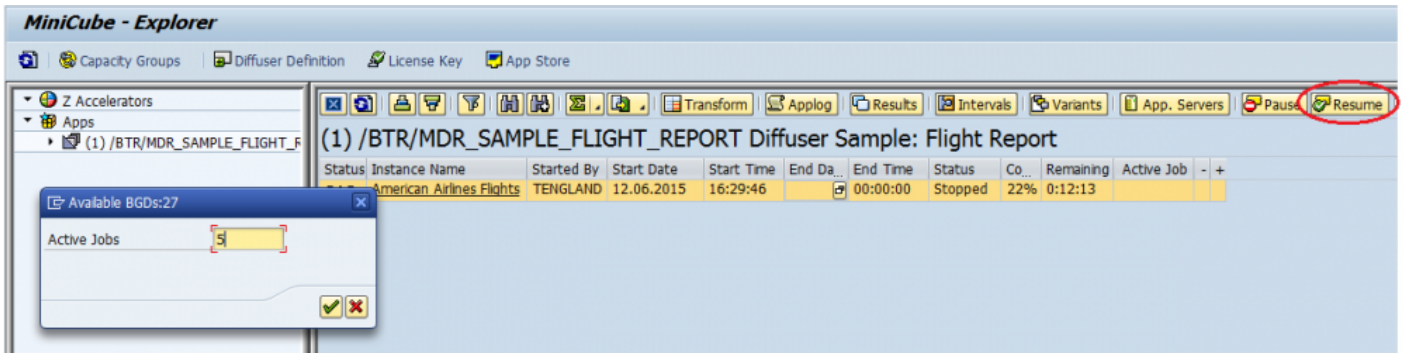
Once all the intervals have completed the status of the instance changes as below.


MiniCube - Explorer
Capacity Groups
Diffuser Definition
License Key
App Store
Z Accelerators
Apps
(1) /BTR/MDR_SAMPLE_FLIGHT_F
Transform
Applog
Results
Intervals
Variants
App. Servers
Pause
Resume
(1) /BTR/MDR_SAMPLE_FLIGHT_REPORT Diffuser Sample: Flight Report
Status
Instance Name
Started By
Start Date
Start Time
End Da...
End Time
Status
Co...
Remaining
Active Job
American Airlines Flights
TENGLAND
12.06.2015
16:29:46
00:00:00
Stopped
22%
0:12:13

Resume

The “Resume” option allows the selected program instance to continue from the point it was stopped or paused. This option uses the Technical Settings of the original program instance to reschedule the report. By resuming an instance it does not reprocess any intervals that have a status of “Completed”, it changes the status of a “Stopped” interval to “Available”. The restart option can only be selected for instances with the status “Paused” or “Error”.

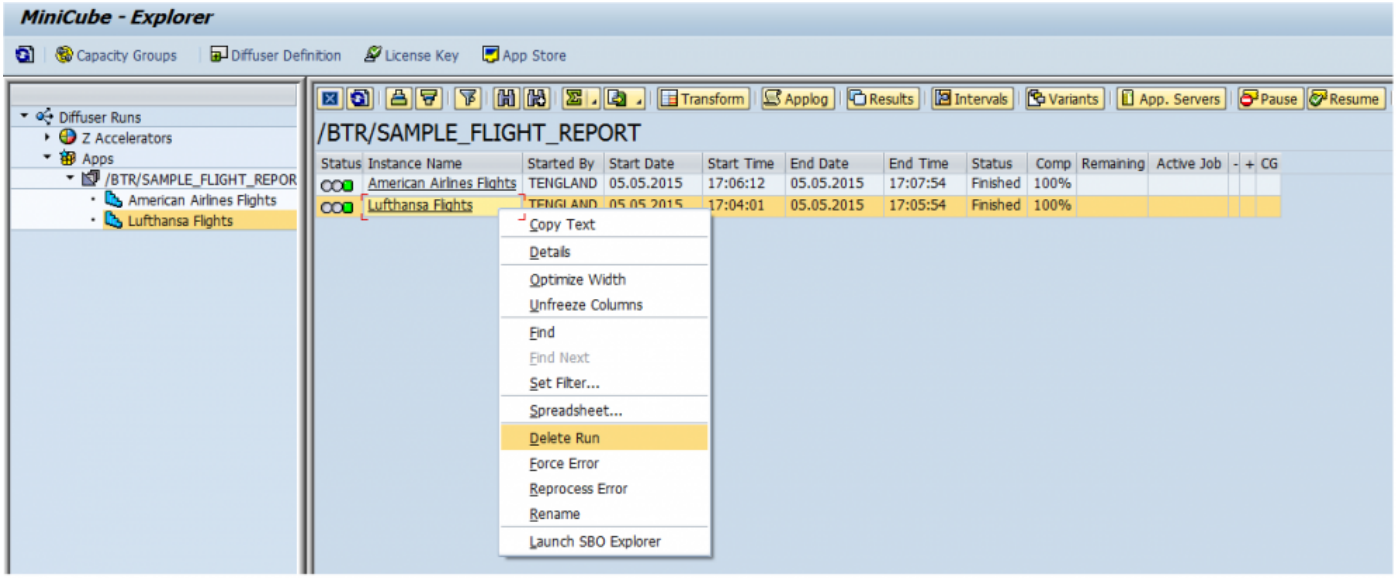
To resume a program select the instance and press the resume button, a pop appears for the number of jobs required to start processing intervals.



Delete

It is possible to delete a program instance by selecting the instance and then the “Delete” option. This in turn deletes all the intervals and results belonging to the program instance. After the delete option is selected the user is faced with a confirmation window to ensure the deletion was intentional. This option is particularly useful in a testing environment and with instances that have errored. It is only possible to delete instances with the status “Error” or “Completed”. The system will not allow an instance “In Progress” to be deleted due to possible data inconsistencies.

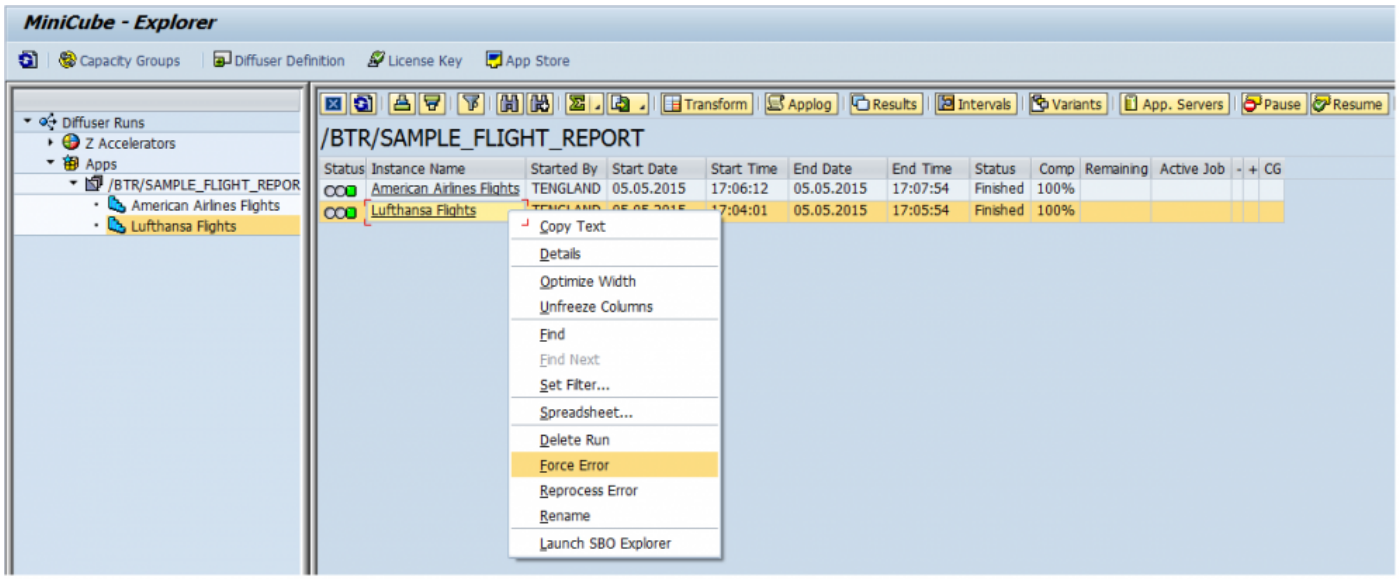
To delete an instance select it right-click and select the Delete option as below.



Force Error

By selecting an Instance, right-clicking and then the “Force error” option, the status of the program instance is changed to “Error”. This allows instances that have technically completed successfully to be changed to Error. This is basically an override function. It is only possible to set an instance to “Error” if there are no active jobs executing the instance.

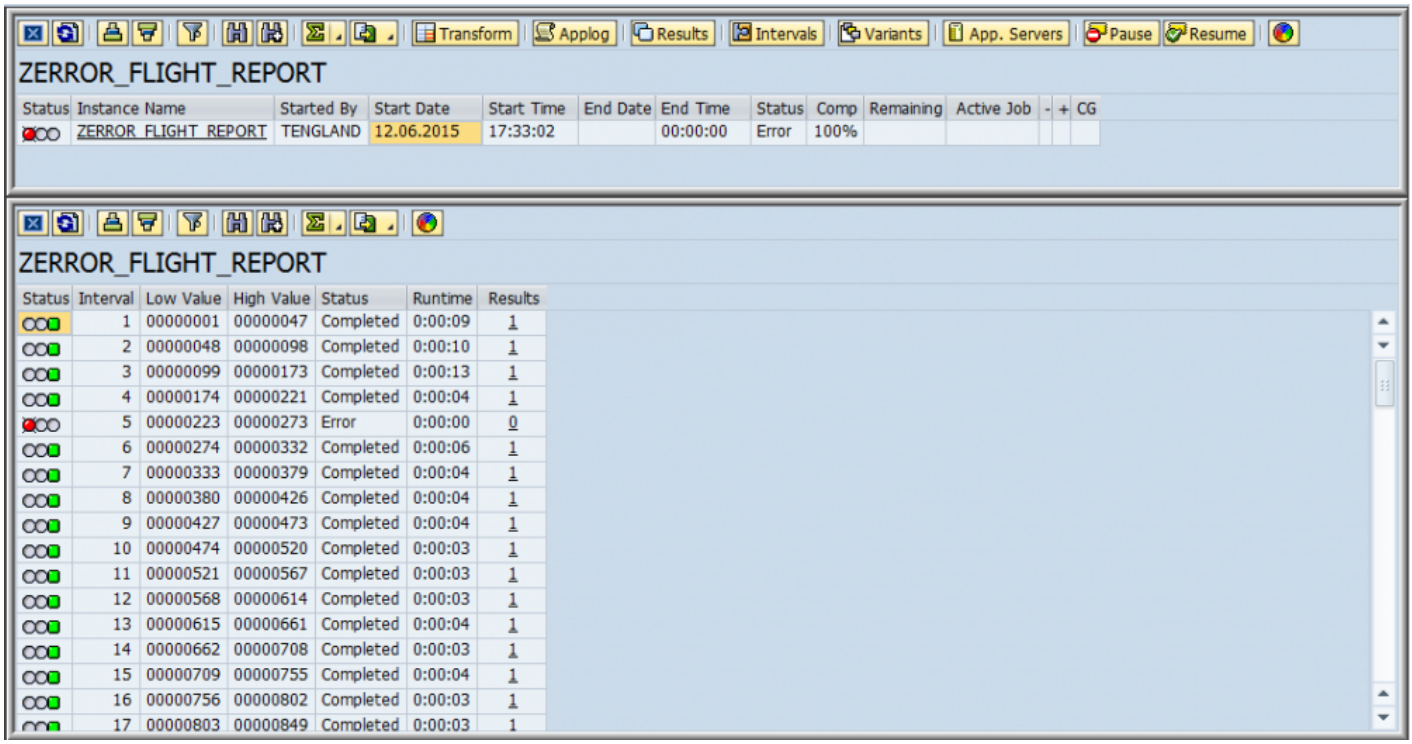
To delete an instance select it right mouse click and select the force error option as below.



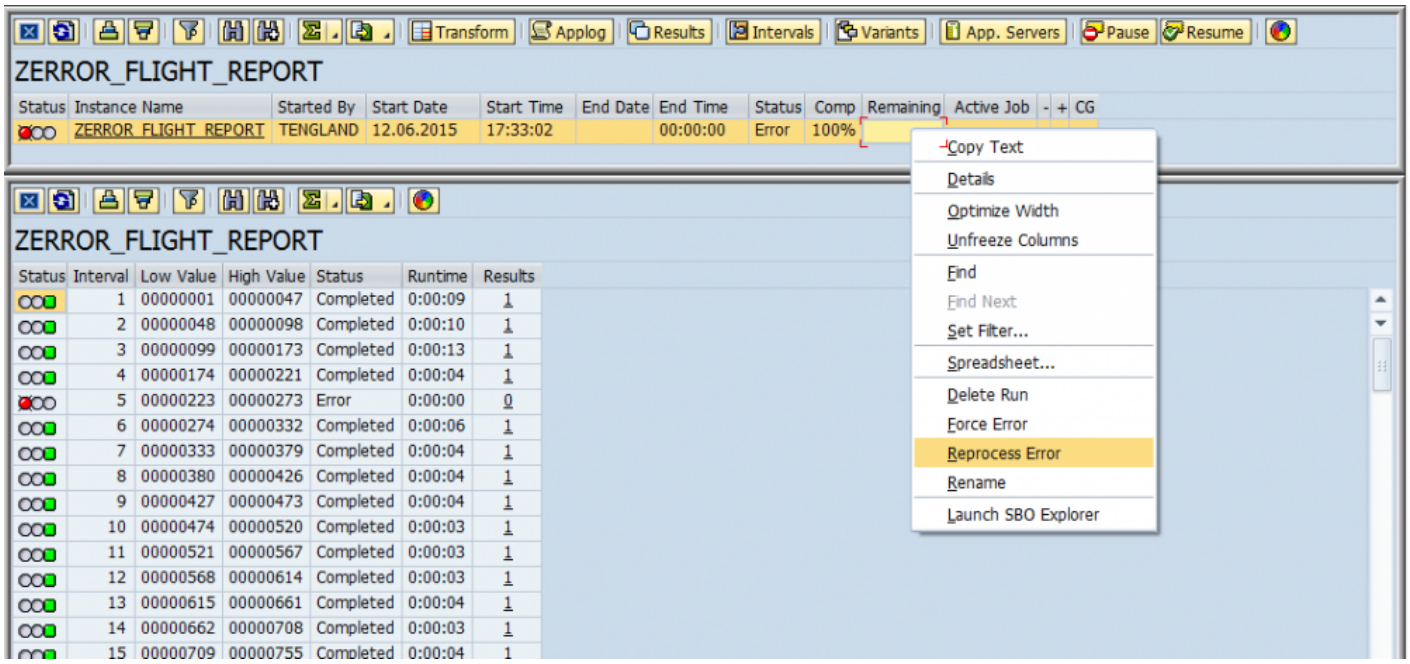
Reprocess Error

On finding an interval in error as below there is an option to reprocess where you have been able to fix the cause of the error, such as updating some master data.

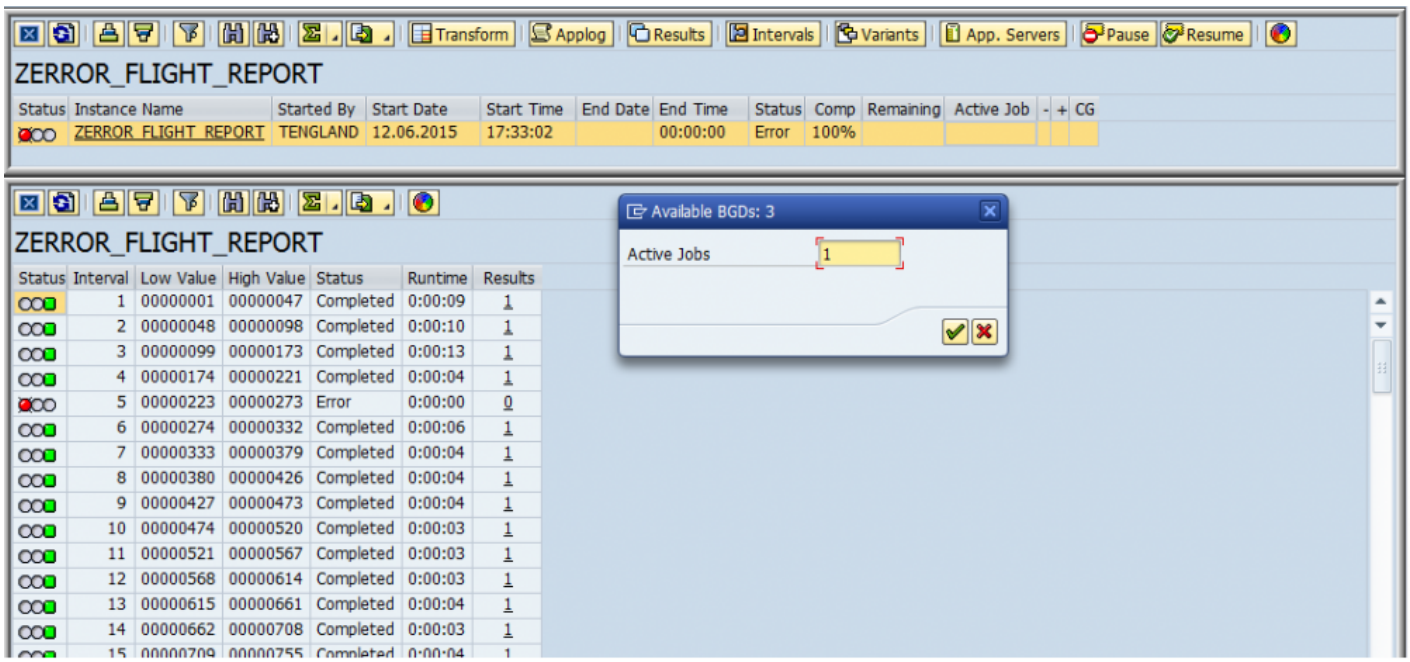
> **!** Bear in mind the impact that running the interval out of sequence or at a later date may have on your report or processing of data.



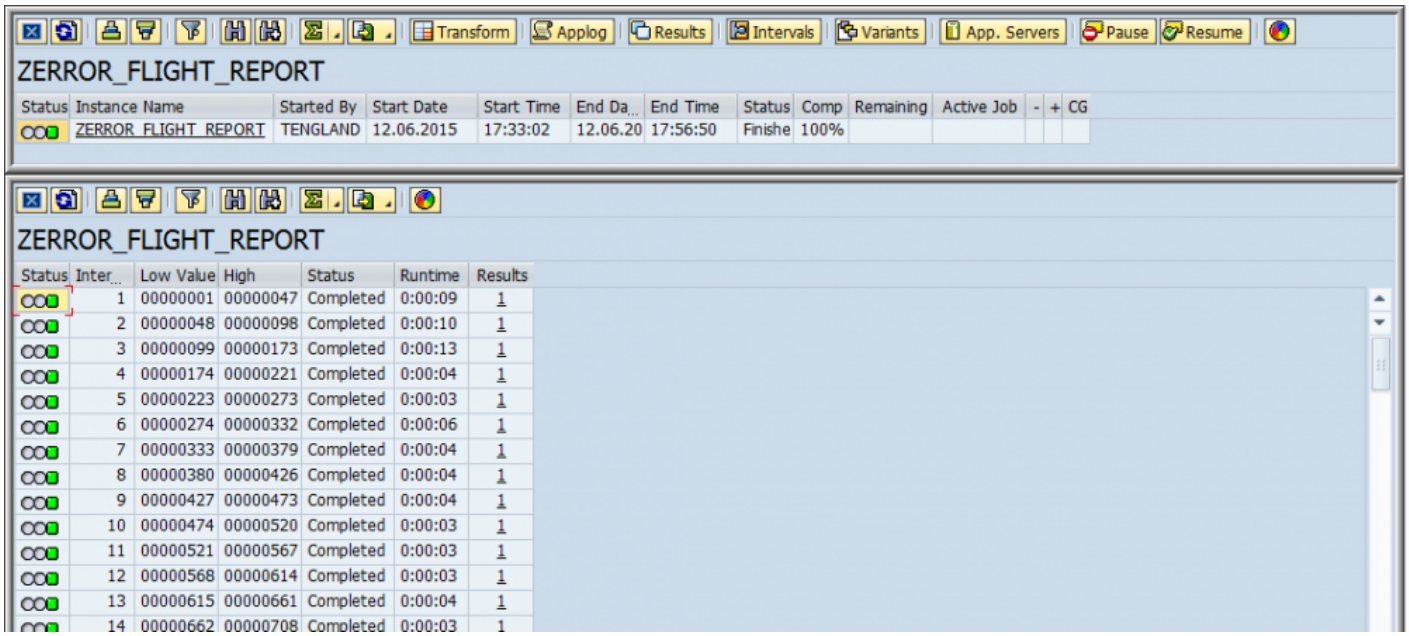
To reprocess the error select the instance in the status of error and right-click for the “Reprocess Error” option as below.



The same as resuming a Diffuser instance the popup for the number of processors you want to utilize appears.



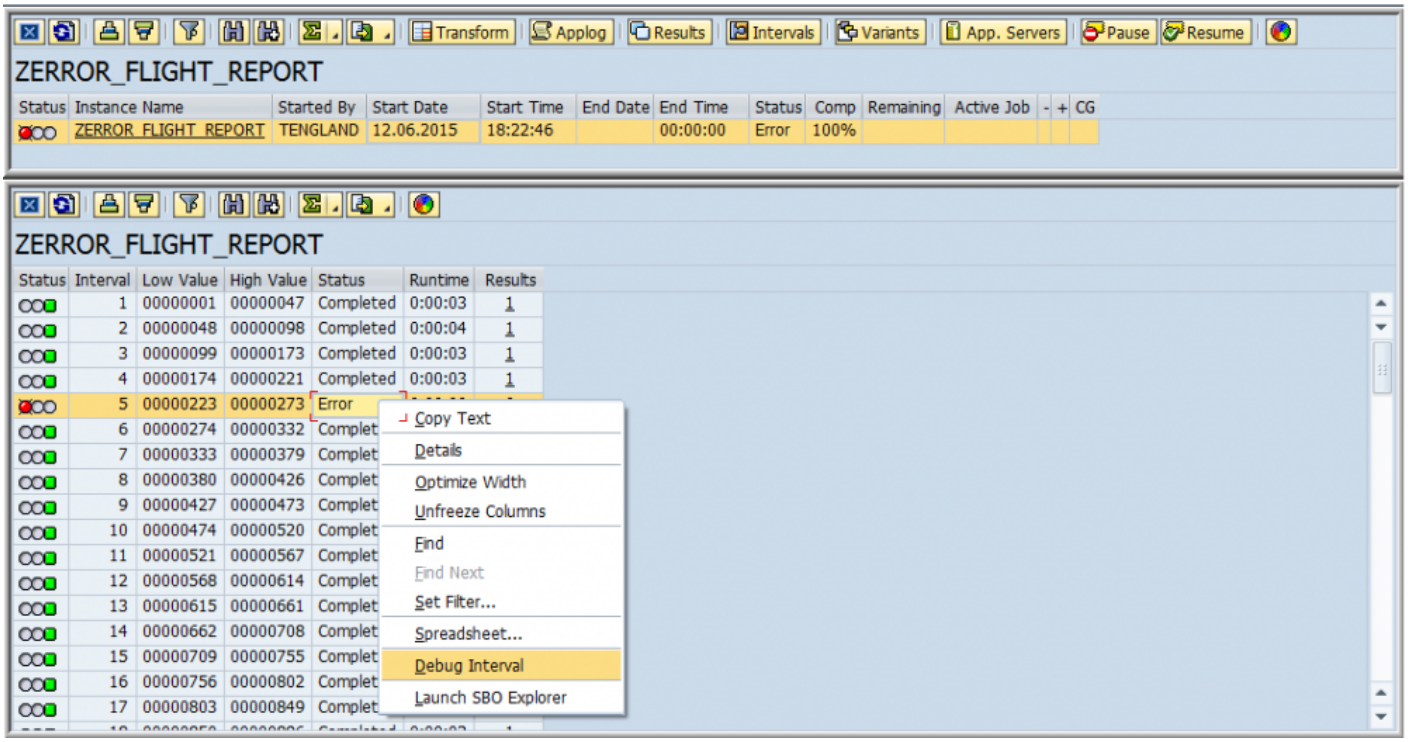
In this example the error is successfully reprocessed.


Transform
Applog
Results
Intervals
Variants
App. Servers
Pause
Resume
ZERROR_FLIGHT_REPORT
Status Instance Name Started By Start Date Start Time End Da... End Time Status Comp Remaining Active Job - + CG
ZERROR_FLIGHT_REPORT TENGLAND 12.06.2015 17:33:02 12.06.20 17:56:50 Finishe 100%
ZERROR_FLIGHT_REPORT
Status Inter... Low Value High Status Runtime Results
1 00000001 00000047 Completed 0:00:09 1
2 00000048 00000098 Completed 0:00:10 1
3 00000099 00000173 Completed 0:00:13 1
4 00000174 00000221 Completed 0:00:04 1
5 00000223 00000273 Completed 0:00:03 1
6 00000274 00000332 Completed 0:00:06 1
7 00000333 00000379 Completed 0:00:04 1
8 00000380 00000426 Completed 0:00:04 1
9 00000427 00000473 Completed 0:00:04 1
10 00000474 00000520 Completed 0:00:03 1
11 00000521 00000567 Completed 0:00:03 1
12 00000568 00000614 Completed 0:00:03 1
13 00000615 00000661 Completed 0:00:04 1
14 00000662 00000708 Completed 0:00:03 1

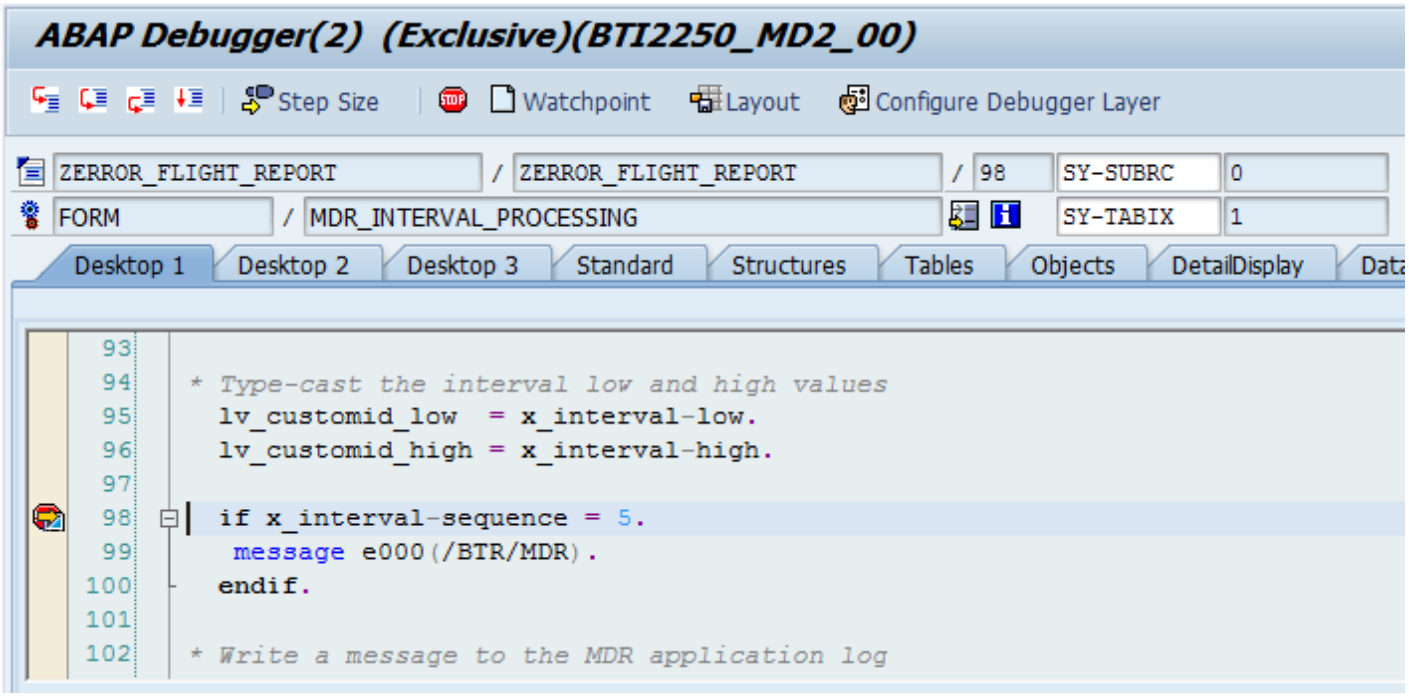
Debug an Interval

On finding an interval in error you also have the option of debugging the interval to try and work out what went wrong.

Firstly ensure you have positioned your break point in the code, then select the interval and right-click for the option to “Debug an Interval”

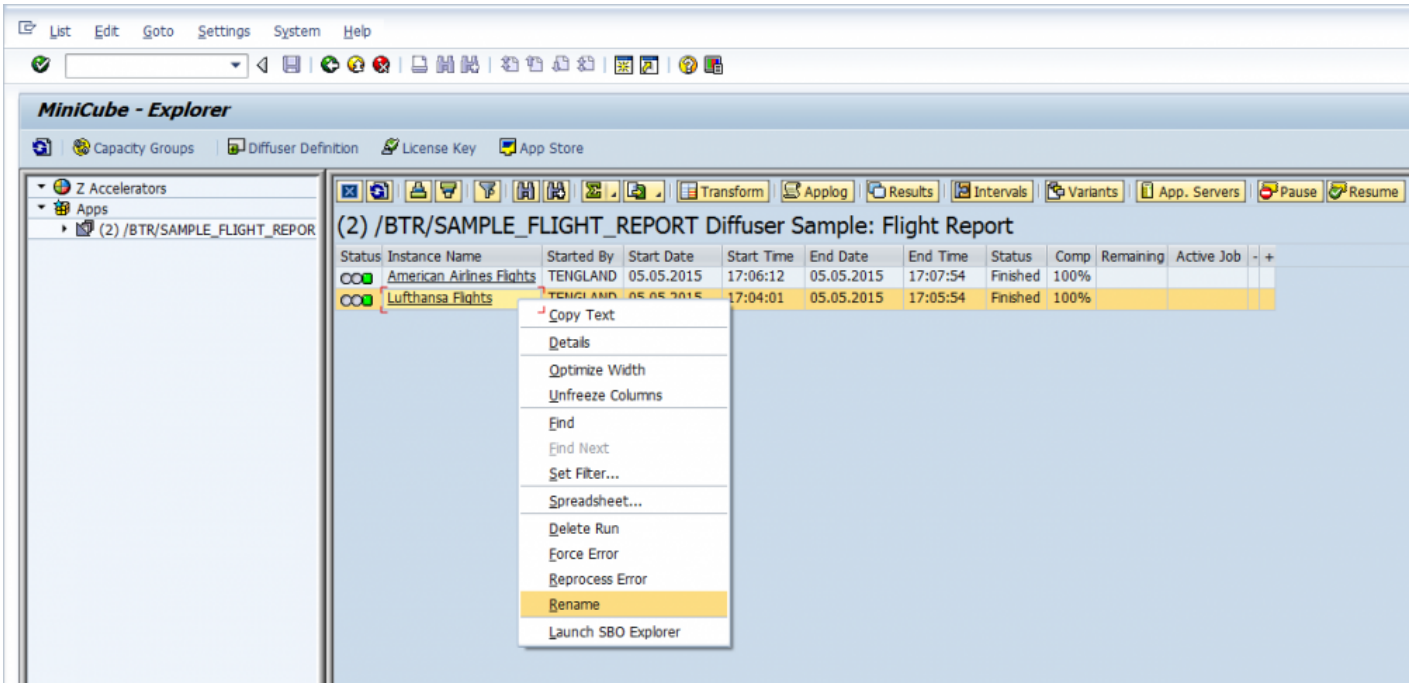


The debugger will then open at your break point.

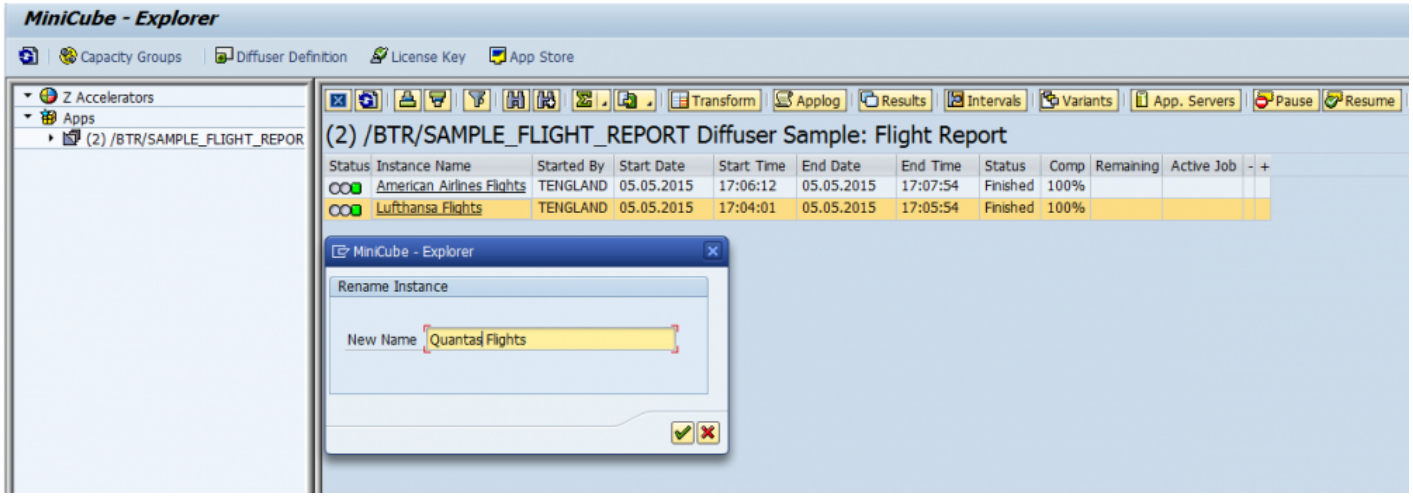

ABAP Debugger(2) (Exclusive)(BTI2250_MD2_00)
Step Size
Watchpoint
Layout
Configure Debugger Layer
ZERROR_FLIGHT_REPORT / ZERROR_FLIGHT_REPORT / 98 SY-SUBRC 0
FORM / MDR_INTERVAL_PROCESSING SY-TABIX 1
Desktop 1 Desktop 2 Desktop 3 Standard Structures Tables Objects DetailDisplay
93
94 * Type-cast the interval low and high values
95 lv_customid_low = x_interval-low.
96 lv_customid_high = x_interval-high.
97
98 if x_interval-sequence = 5.
99 message e000(/BTR/MDR).
100 endif.
101
102 * Write a message to the MDR application log

Rename Instance

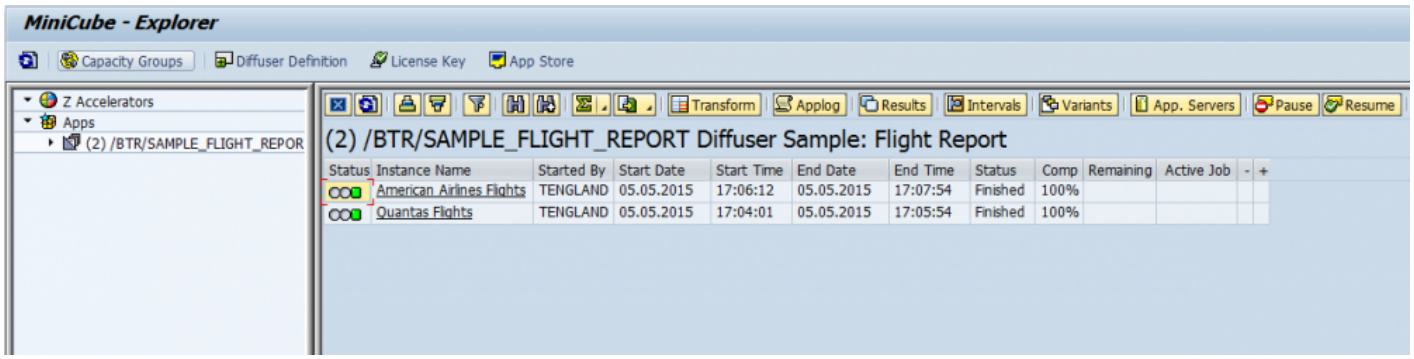
To rename an instance select the instance and right-click.



Enter the new name.



The new name is updated as below:


MiniCube - Explorer
Capacity Groups
Diffuser Definition
License Key
App Store
Z Accelerators
Apps
(2) /BTR/SAMPLE_FLIGHT_REPOR
Transform
Applog
Results
Intervals
Variants
App. Servers
Pause
Resume
(2) /BTR/SAMPLE_FLIGHT_REPORT Diffuser Sample: Flight Report
Status Instance Name Started By Start Date Start Time End Date End Time Status Comp Remaining Active Job
American Airlines Flights TENGLAND 05.05.2015 17:06:12 05.05.2015 17:07:54 Finished 100%
Quantas Flights TENGLAND 05.05.2015 17:04:01 05.05.2015 17:05:54 Finished 100%

Scheduling Diffuser Programs

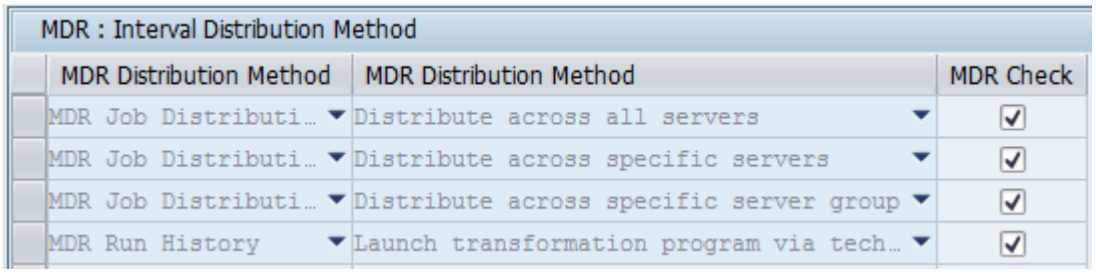
A Diffuser program can be scheduled just like any other background program. Typically this is done using the standard transaction SM36. The program variants can also be saved as per normal.

Node5 Diffuser in most cases, however, does require another program to be scheduled for it to operate efficiently in a production environment. The program function is to regenerate the Interval Variant. The purpose of regenerating an Interval Variant is such that as the master or transactional data grows, the intervals can be recalculated to ensure that each interval is evenly spread. This then ensures the Diffuser program is processed as efficiently as possible.

The program /BTR/MDR_INTERVAL_REGENERATION is used for this purpose. This job should typically be scheduled nightly at the beginning of the batch window, and can be executed for individual Interval Objects, individual Interval Variants, or for all Interval Variants by adjusting the parameters on the selection screen. For the Interval Regeneration to operate, you will need to configure the table /BTR/INTVALVARC. Here you define an Interval Object, Interval Variant and the refresh age. The refresh age defines how frequently the Interval Variant is refreshed. For example if for Object SCUSTOMID, Variant SAMPLE, if the refresh age is 7, the interval variant will only be regenerated every 7 days, even if the regeneration job is scheduled nightly. This functionality allows you to avoid scheduling individual regeneration jobs in different reoccurring cycles. You can override the refresh age functionality by selecting the “Force regeneration” check-box.

Job Distribution

Diffuser programs can now be run on a specific server/servers or a server group. This functionality can be activated on table /BTR/MDR_C via transaction code SM30 as shown below.

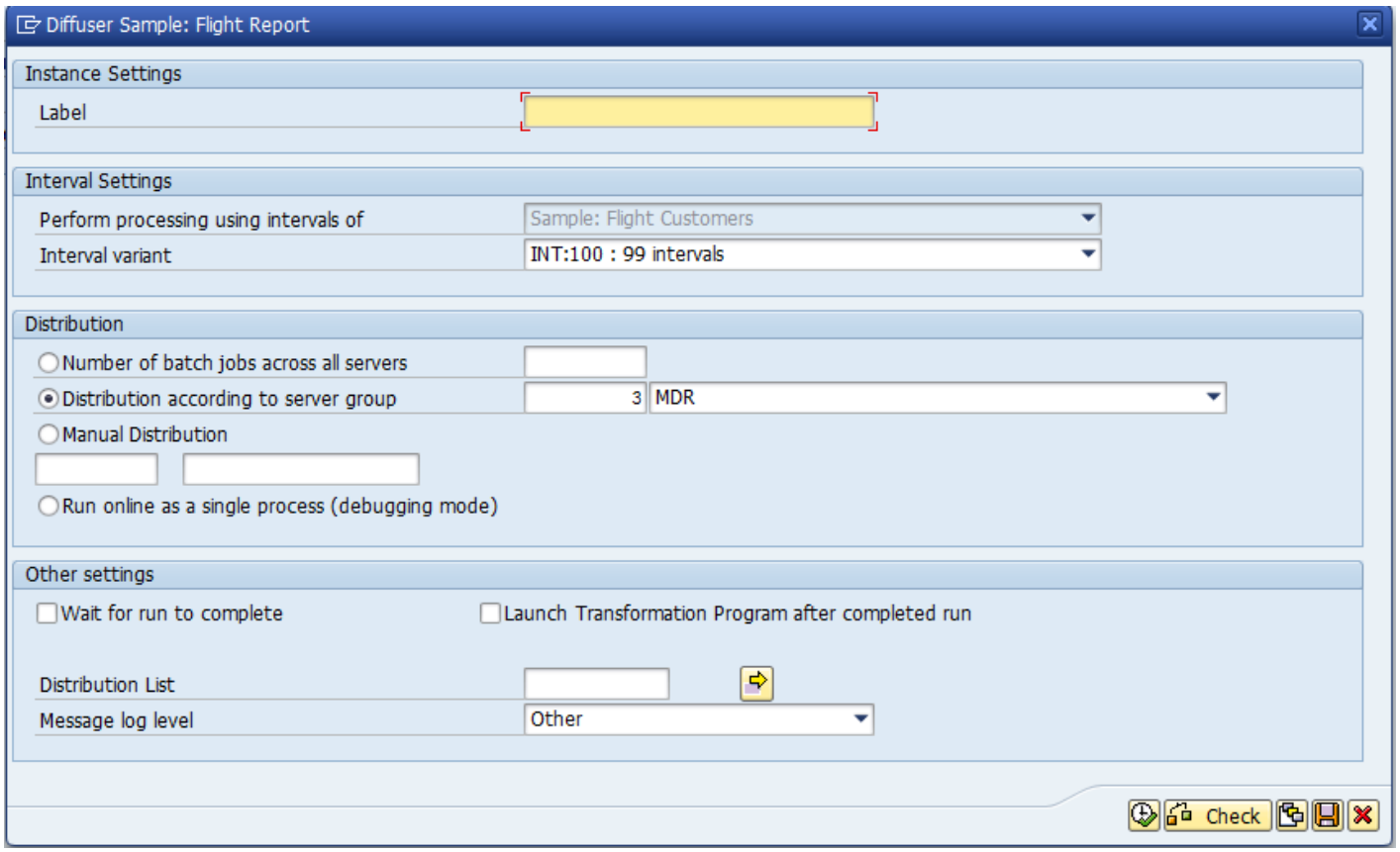


- Server Group Distribution
- Manual Distribution

Server Group Distribution

If option “Distribute across specific server group” in config table /BTR/MDR_C is checked an additional option will appear on the technical settings screen.

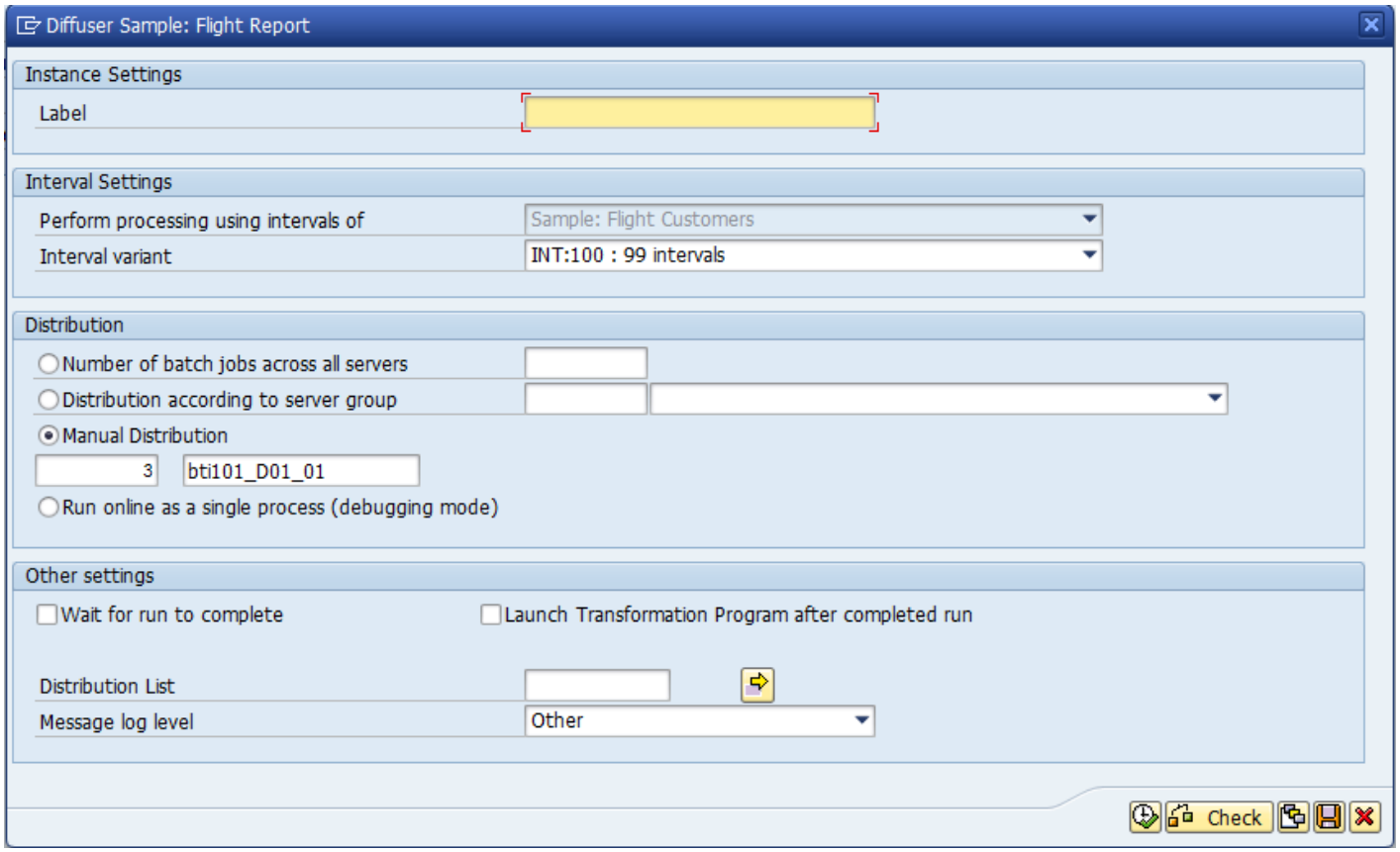
Specify the number of jobs and predefined server group in the corresponding input fields (see below) and run your Diffuser program as usual.



Manual Distribution

If option “Distribute across specific servers” in config table /BTR/MDR_C is checked an additional option will appear on the technical settings screen. The Node5 Diffuser determines how many application servers are available (up to 10 servers) and displays the corresponding number of input field rows (in the example below only one).

Specify the number of jobs as well as application server in the corresponding input fields by means of the search help and run your Diffuser program as usual (see below).



Maintenance of MiniCube results

The Node5 Architecture provides program /BTR/MDR_INSTANCE_DELETE, this program allows you to completely delete the stored result sets or MiniCubes of data for one or more Diffuser instance runs and the result sets of all their corresponding intervals. Alternatively, you can choose to delete only the result sets of the intervals of an instance run. If you don't need results at interval level you can choose the second options since the result sets at instance level are an aggregation of the intervals.

To run the program /BTR/MDR_INSTANCE_DELETE you need to have added the programs you want to delete the data for in the table /BTR/PROGARCH using SM30. This allows you to restrict by program and numbers of days of instance runs that are stored, the configuration table makes it difficult for anyone to accidentally delete results. For example a sales team might want to see their historical data for the last 30 days and you can add 30 as the limit, then when running the /BTR/MDR_INSTANCE_DELETE it will only delete data for that program that is older than the 30 day limit.

It is recommended to schedule this program to do regular cleaning in your system to avoid accumulation of unnecessary historical data.

If you only need to delete one run the "Delete" functionality for individual instances mentioned in Administering Diffuser Programs can be useful.

Performance considerations

The limitation of Node 5 Diffuser is purely based upon the available hardware within the SAP landscape. We have seen, on larger SAP customers, 20+ application servers with over 700 background processes. In this configuration, it has been possible to run a Diffuser program with as much as 300 parallel processes without causing contention upon the database and seeing an effect on scalability.

Contention with other batch jobs and dialog processes

It is important to know what other jobs are running in the batch schedule so that the SAP system is not overloaded. This goes for both Diffuser programs and their child processes as well as other batch jobs or user / dialog activity. This is the key benefit of a batch scheduler to ensure that jobs are orchestrated together. Node5 Diffuser arms you with the necessary control to run an ABAP program in a short burst of activity, so you can get it out of the way by maximizing the available hardware and system configuration.

Diffuser and your SAP batch scheduler

Node5 Diffuser integrates seamlessly with any job scheduler e.g. Automic / UC4, Tivoli / Maestro, Redwood. The scheduler itself triggers the same Diffuser ABAP program with the same variant. Within this variant, we set a parameter flag (see Wait for run to complete in section 1.2) that ensures that the “parent” job (triggered by the job scheduler) waits while the child parallel jobs finish. This means there are no changes from a batch scheduler perspective, as it completes the job as per normal but just N times faster. One of the key strengths of a Diffuser report / program is that it looks like any other ABAP report / program. The number of “intervals” and the number of “parallel jobs” to be started must be specified, however, these can be defaulted into the program itself or into the variant. The batch scheduler will still continue scheduling the program to run with the ABAP program name and the variant name, but the parameters within that variant will be enhanced only.

Security Enhancements

As mentioned above, the Defaults for Technical Settings section offers two options for functionality restrictions, “Lock Technical Settings” and “Lock Expert Mode”. These work at a program level and once set they will apply for every user.

However, the Node5 Architechture also provides enhancement spots to allow developers to apply customer specific authority checks. This can be used to restrict technical as well as administrative settings at user and at program level.

For more information refer to the section Authority Checks in the Z Accelerators Guide.

Software Support

Online Forum

Basis Technologies have an online forum containing over 250 searchable Frequently Asked Questions relating to our products.

These FAQs cover many of the common error / warning messages that can be experienced during normal usage and also useful HOW TO guides to perform many of the common operations.

The online forum can be accessed via the following URL:

http://support.basistechnologies.com/forums

You will need to register for a username and password before you can access the forum.

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.