# Operator Combination and Control

## Introduction

Orchestrate® Shell (OSH), a scripting language used to create a parallel job application, is integrated with the DataStage® Suite as “Parallel Extender” (now called IBM® InfoSphere™ DataStage Enterprise Edition). As “Operator” is an Orchestrate terminology, we can see the term “operator” widely used in the DataStage application. The term “Stage” used in DataStage designer module is an analogy for operator.

An Operator, used to process the data, is the basic fundamental unit underlying any IBM InfoSphere DataStage job. Various stages are available in a DataStage job. These stages are based on the operators that will analyze and process the data, in the data flow that exists in the job design.

An operator or stage reads input data sets, does necessary processing, and delivers datasets as an output. Users design (through graphical user interface) the job using the tool palette in the DataStage designer module. Once they compile the job, an OSH script is created. This script has all the operators and their input, and output datasets that need to be executed during the job run.

## Execution of Job / OSH Script

When a DataStage job is executed, it refers the configuration file that describes the nodes and resource available in the server to run the job. When a job runs, a sequence of steps occurs initiating a conductor node, section leaders and players. See Figure 1.

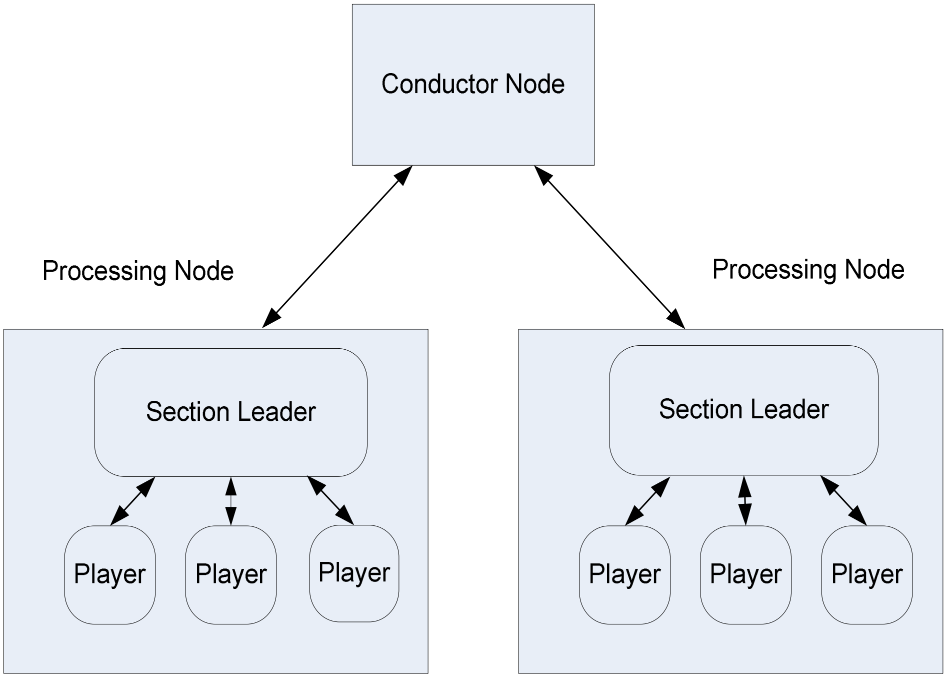


**Figure 1: Orchestrate Architecture**

### Conductor Node

Based on a configuration file, the conductor node determines the number of instances of each operator to be run. It will start one section leader per processing node. If child processes fail, the error message is passed as an output. The conductor node will also take care of cleaning files and proper shut down of the process in the event of failure.

### Section Leader

In every processing node, a section leader will be created. The operators are run on this node, by creating players. The section leader will communicate with the conductor node and players. The section leader will also take care of cleaning files and provide proper shut down of the process in the event of failure. When failure occurs, the section leader also reports error messages to the conductor node.

### Players

Players are the actual work-horse of the operators. At least one player will be created for one operator in a processing node. The section leader may start more than one player – when this occurs, multiple players combine and run as a single process by default, to improve performance. Players combining as a single process is termed, “Operator Combination”.

Players can communicate with other players in the data flow. Players also take care of housekeeping upon process completion. Housekeeping includes cleaning the temporary files created while processing the records.

## Operator Combination

Operator combination is the default behavior of DataStage for job optimization process. Operators send requests to nearby operators to combine. When the request is accepted, the operators will be combined and run as the same process. Operators (or players – i.e., instances of operator) combining must be in the same processing node and with the same degree of parallelism. See Figure 2.

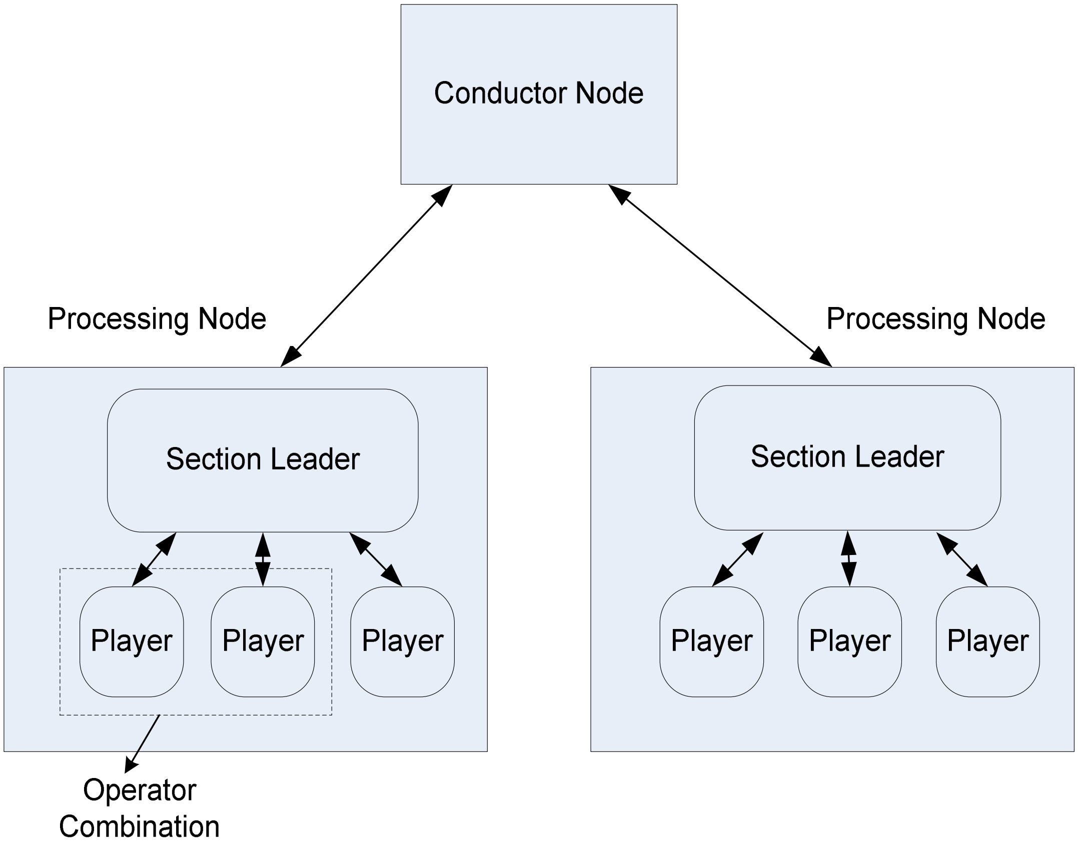


**Figure 2: Orchestrate Architecture Depicting Operator Combination**

## Operator Combination – Advantages

The advantages of operator combination are:

- In case of two operators running separately as two different processes, the operators have to transit records between them. When these two operators combine, the records transit time is reduced.
- In case of two operators running separately as two different processes, the operators take the input data sets and send the output data sets via a buffer that has limitations (32 kilobyte). But when the operators combine there are no such restrictions.
- Operator combination minimizes shared memory and resource usage. It also lessens the inter-process communication.
- Operator Combination intends to separate I/O from the CPU activity.

## Operator Combination – Disadvantages

There are also disadvantages of operator combination, including:

- Diagnosing the reason behind the occurrence of problems and errors, in the event of DataStage job failure is difficult.
- Operator combination might slow down the performance by reducing the pipeline parallelism, as they share one thread of control.
- Operator combination does not always improve performance, and therefore, it is not suited for all applications. In some cases, it will expose the complexity of internal API.
- Disabling operator combination generates more UNIX processes to run the job – The job may finish faster or slower depending on the server sustainability with more processes.
- It allows each operator to use more CPU (Use DataStage job monitor to identify the CPU bottlenecks).

## Operator Combination Enabled / Disabled - Comparison

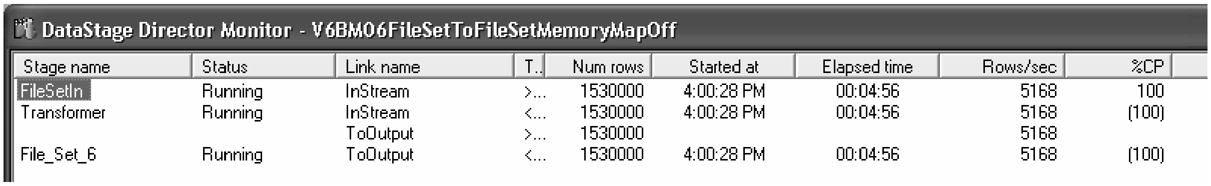
### *With Operator Combined.*

(Below are parts of a job log)

```
It has 2 operators:
op0[1p] {(parallel APT_CombinedOperatorController:
(FileSetIn.InStream)
(APT_TransformOperatorImplJob_Transformer in Transformer)
(APT_RealFileExportOperator in File_Set_6.ToOutput)
) on nodes (
node1[op0,p0]
)}
op1[1p] {(sequential APT_WriteFilesetExportOperator in File_Set_6.ToOutput)
on nodes (
node1[op1,p0]
)}
```

**It runs 2 processes on 1 node.**

- APT_CombinedOperatorController indicates operator combination occurred. FileSet and Transformer operators are combined.
- Note in Job Monitor, the %CP is 100. One process cannot use more than 100% of CPU. So if operator combination is disabled, the operator uses more CPU resources and makes the job finish faster.



DataStage Director Monitor - V6BM06FileSetToFileSetMemoryMapOff

| Stage name | Status | Link name | T.. | Num rows | Started at | Elapsed time | Rows/sec | %CP |
|---|---|---|---|---|---|---|---|---|
| FileSetIn | Running | InStream | >... | 1530000 | 4:00:28 PM | 00:04:56 | 5168 | 100 |
| Transformer | Running | InStream | <... | 1530000 | 4:00:28 PM | 00:04:56 | 5168 | (100) |
|  |  | ToOutput | >... | 1530000 |  |  | 5168 |  |
| File_Set_6 | Running | ToOutput | <... | 1530000 | 4:00:28 PM | 00:04:56 | 5168 | (100) |

### *Without Operator Combined.*

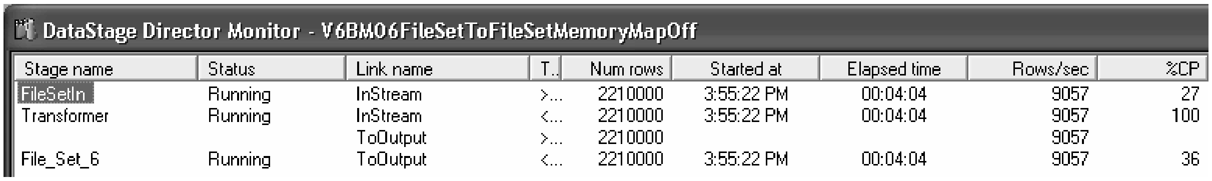
(Below are parts of a job log)

```
It has 4 operators:
op0[1p] {(parallel FileSetIn.InStream)
on nodes (
node1[op0,p0]
)}
op1[1p] {(parallel APT_TransformOperatorImplJob_Transformer in Transformer)
on nodes (
node1[op1,p0]
)}
```

```
op2[1p] {(parallel APT_RealFileExportOperator in File_Set_6.ToOutput)
on nodes (
node1[op2,p0]
)}
op3[1p] {(sequential APT_WriteFilesetExportOperator in File_Set_6.ToOutput)
on nodes (
node1[op3,p0]
)}
```

**It runs 4 processes on 1 node.**

- No APT_CombinedOperatorController appears in the log, so no operator combination occurred. All the operators run separately.
- Note in Job Monitor the %CP; the CPU utilization is more.



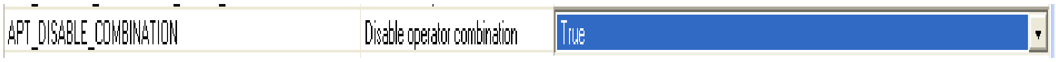
DataStage Director Monitor - V6BM06FileSetToFileSetMemoryMapOff

| Stage name | Status | Link name | T.. | Num rows | Started at | Elapsed time | Rows/sec | %CP |
|---|---|---|---|---|---|---|---|---|
| FileSetIn | Running | InStream | >... | 2210000 | 3:55:22 PM | 00:04:04 | 9057 | 27 |
| Transformer | Running | InStream | <... | 2210000 | 3:55:22 PM | 00:04:04 | 9057 | 100 |
|  |  | ToOutput | >... | 2210000 |  |  | 9057 |  |
| File_Set_6 | Running | ToOutput | <... | 2210000 | 3:55:22 PM | 00:04:04 | 9057 | 36 |

## Disable Operator Combination

At times, operator combination may need to be disabled. This section includes information on how to disable an operator combination.

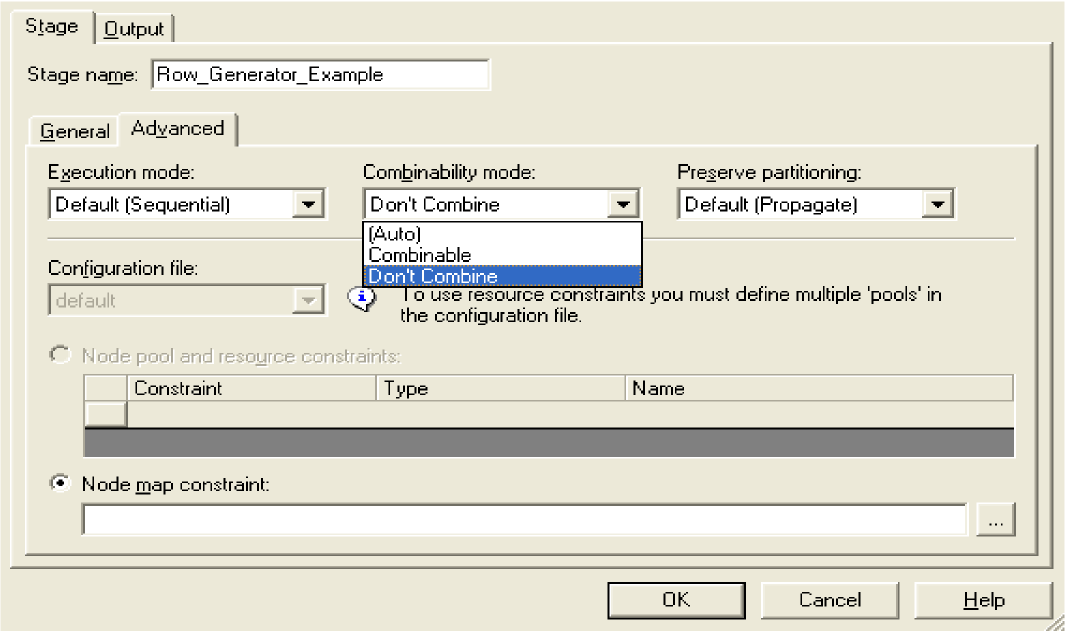
The environmental variable $APT_DISABLE_COMBINATION, set to true, will disable operator combination. It will globally disable operator combination, create more processes in the server and impact performance significantly.



|  |  |  |
|---|---|---|
| APT_DISABLE_COMBINATION | Disable operator combination | True |

Before disabling operator combination by setting the environmental variable to true, analyze the job score to find whether the operator combination occurred. If you see "APT_CombinedOperatorController" in the job score, then it denotes operator combination has been occurring. In case of non-combining of operators and disabling (setting true) the environmental variable globally has no impact with the current working job, but it might have effect on other jobs that have an operator combination process.

Operator combination could also be avoided by options available in the stage properties within the DataStage designer. See Figure 3.



Stage name: Row_Generator_Example

Execution mode: Default (Sequential)

Combinability mode: Don't Combine

(Auto)
Combinable
Don't Combine

Preserve partitioning: Default (Propagate)

Configuration file: default

To use resource constraints you must define multiple 'pools' in the configuration file.

○ Node pool and resource constraints:

| Constraint | Type | Name |
|---|---|---|
| | | |

● Node map constraint:

**Figure 3: Combinability Mode in DataStage Stage Property**

In the combinability mode if “Don’t Combine” option is selected, then this stage / operator will avoid combining with a nearby operator.

## Operator Combination Controller

All combinable operators are handled by the combined operator controller. One combined operator controller will be run for a group of combinable operators in a processing node. This combined operator controller is assigned by the parallel framework itself. The combined operators process the records and pass them to the controller. This controller passes the output records from the combined operators, as input to the next operators existing in the DataStage job design data flow.

## Conclusion

Operator combination might improve the job performance, but in a few cases it will slow down the process. Based on your server configuration, enable or disable operator combination to boost the performance.

*PR3 Systems helps their clients to extract critical and strategic information regarding their business from islands of scattered data within their organization. Their services include training, consulting, hardware, software, and strategic road mapping to provide a scalable cost-effective complete end-to-end solution to achieve client goals. For more information on this topic, or to contact us regarding our DataStage consulting and training services: call us at 630-364-1469, email us at info@pr3systems.com or visit our web site at www.pr3systems.com.*

[®] Orchestrate, DataStage, and IBM are registered trademarks of International Business Machines Corporation (IBM Corporation).

[™] InfoSphere is a trademark of International Business Machines Corportion (IBM Corporation).