
Ab inito Session 10
Introduction to Ab Initio
Capgemini
Ab Initio Training
1

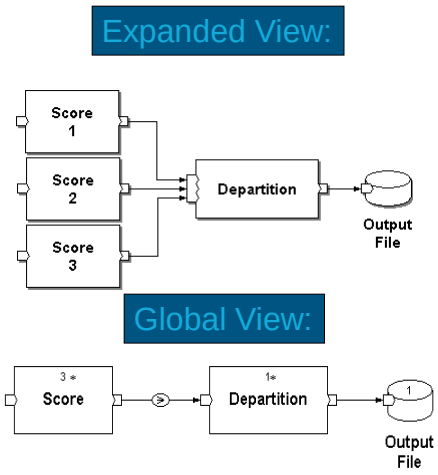
De-Partitioning Components

- **Gather**
- **Merge**
- **Concatenate**
- **Interleave**

De-Partitioning

- Departition components combine multiple streams of input. Usually a Departition component is preceded by a fan-in flow.
- To reverse the effect of a Departition component, use a Partition component.
- Departitioning combines many flows of data to produce one flow.
- Each departition component combines flows in a different manner.

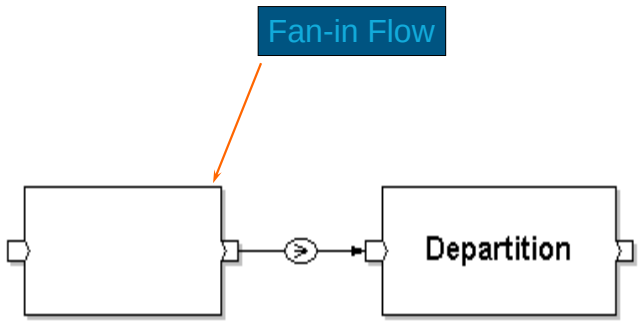
Departitioning



CapGemini Ab Initio Training

Departitioning

- For the various departitioning components:
  - Key-based?
  - Result ordering?
  - Effect on parallelism?
  - Uses?



CapGemini Ab Initio Training

Gather

- Gather combines data records from multiple flow partitions (mfs) arbitrarily and make the flow serial and collect from different serial flow of same type (of same dml) to make it single flow.

Gather

Round-robin partitioned and scored:

```
42John 02116 30A
45Sue  02241 92A
48Mary 02116 38A
```

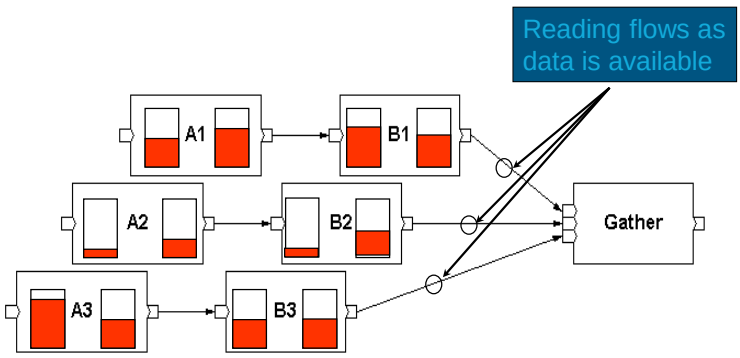
```
43Mark 02114  9C
46Rick 02116 23B
49Jane 02241  2C
```

```
44Bob  02116  8C
47Bill 02114 14B
```

Scored dataset in random order, following gather:

```
43Mark 02114  9C
46Rick 02116 23B
42John 02116 30A
45Sue  02241 92A
48Mary 02116 38A
44Bob  02116  8C
47Bill 02114 14B
49Jane 02241  2C
```

Gather: Performance



Note that the Gather will not affect upstream processing

Gather

- Not key-based.
- Result ordering is unpredictable.
- Has no affect on the upstream processing.
- Most useful method for efficient collection of data from multiple partitions and for repartitioning.
- Used most frequently

Merge

- Merge combines data records from multiple flow partitions that have been sorted according to the same key specifier (for reference see yellow mark in parameter box), and maintains the sort order.
- CAUTION: While using merge component use flow buffering.

Merge

Round-robin partitioned and sorted by amount:

```
42John 02116 30
48Mary 02116 38
45Sue  02241 92
```

```
49Jane 02241  2
43Mark 02114  9
46Rick 02116 23
```

```
44Bob  02116  8
47Bill 02114 14
```

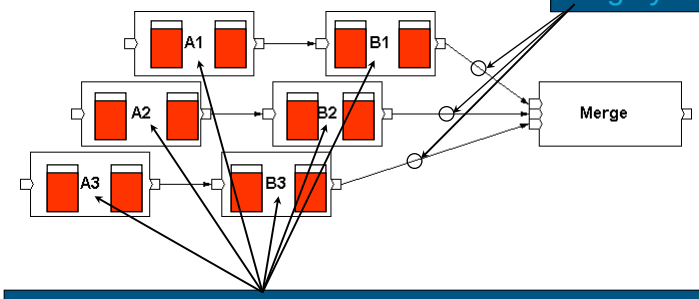
Sorted data, following merge on amount:

```
49Jane 02241  2
44Bob  02116  8
43Mark 02114  9
47Bill 02114 14
46Rick 02116 23
42John 02116 30
48Mary 02116 38
45Sue  02241 92
```

Merge: Performance

If keys evenly distributed:

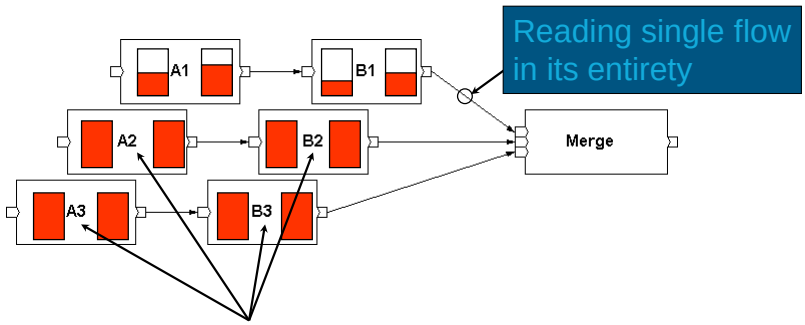
Reading flows roughly evenly



Components running roughly in lock-step, thus computation becomes synchronized across partitions

Merge: Performance

If keys globally sorted or near globally sorted:



Blocked components, thus computation becomes serialized across partitions

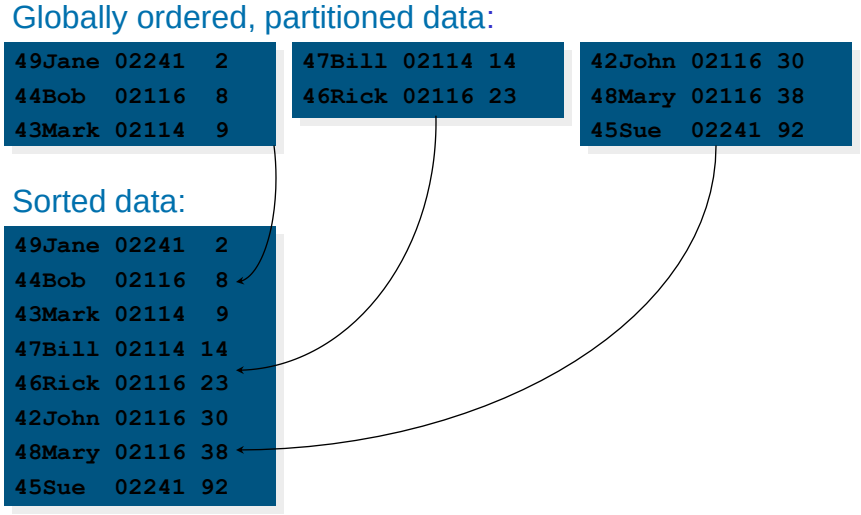
Merge

- Key-based.
- Result ordering is sorted if each input is sorted.
- Possibly synchronizes pipelined computation; may even serialize.
- Useful for creating ordered data flows.
- Other than the 'Gather', the Merge is the other 'departitioner' of choice

Concatenate

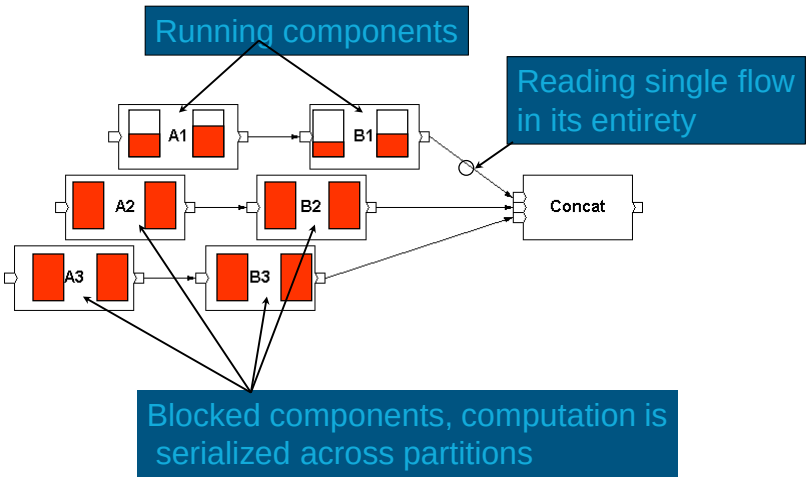
- Concatenate appends multiple flow partitions of data records one after another.
- Example : If the requirement is to generate and output file containing header, body and trailer part (all parts are from different flow)

Concatenate



CapGemini Ab Initio Training 116

Concatenate: Performance

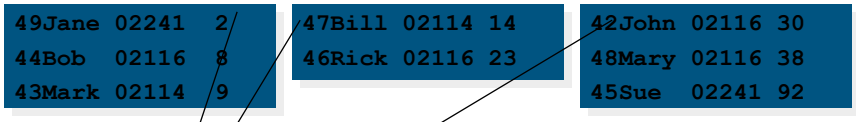


Concatenate

- Not key-based.
- Result ordering is by partition.
- Serializes pipelined computation.
- Useful for:
  - appending headers and trailers
  - creating serial flow from partitioned data
- Used very infrequently

Interleave

Globally ordered, partitioned data:



Departioned data:

```
49Jane 02241 2
47Bill  02114 14
42John 02116 30
44BOb 02116 8
46Rick 02116 23
48Mary 02116 38
43Mark 02114 9
45Sue  02241 92
```

INTERLEAVE Cont.

- PARAMETERS
- Blocksize:-Number of data records interleave reads from each flow before reading the same number of data records from the next flow.

The Interleave component:

- Reads the number of data records specified in the blocksize parameter from the first flow connected to the in port
- Then reads the number of data records specified in the blocksize parameter from the next flow, and so on
- Writes the records to the out port

Summary of Departitioning Methods

Method	Key-based?	Ordering?	Uses
Gather	No	Unpredictable	Unordered departitioning, repartitioning
Merge	Yes	Sorted	Creating ordered serial flow
Concatenate	No	Global	Creating serial flow in partition order
Interleave	No	Global	Creating serial flow in round-robin fashion

Differences between DePartitioning Components

- There are some basic difference of concatenate, gather and merge which are mentioned as below
- Concatenate: Append different flows of same types (same dml) in order in a single flow.
- In the example, concatenate will always take header record, then detail and then trailer.
- Gather: Collect different flow arbitrarily.
- Merge: Collect different flows and maintain the sorted order.
- But the gather will do this arbitrarily

Thank You

End of Session 12