

Benchmark: Talend Open Studio vs Pentaho Data Integrator (aka Kettle)

V0.23

MarcRussel@gmail.com
Last modified: 2007-07-31

Table of contents

Environment.....	2
Test 1: Text Input file > Text Output file.....	3
Test 2: Text Input file > XML Output file.....	5
Test 3: Text Input file > Mysql Output table.....	7
Test 4: Text Input file > Transformation > Text Output file.....	9
Test 5: Test 4 + Lookup.....	11
Test 6: Test 5 + output filter.....	13
Test 7: Test 6 + aggregation.....	15
Appendix 1: Transformation step/component.....	17

Environment

Comparison benchmarks were performed on **TOS 2.1.0RC1** and **TOS 2.1.r4725** vs **PDI/Kettle 2.5.0** and **PDI 3.0.0M1**.

TOS 2.1.r4725 & PDI 3.0.0M1 have shown global enhancements as far as performance is concerned.

Some test results are missing on PDI 3.0.0M1, due to a component functional bug, preventing some of the tests to run properly.

Tests were carried out using files of 10,000 lines, 100,000 lines and 5 million lines.

Tests with 10,000 and 100,000 records were executed 4 times – best result of four was retained, whereas, tests with 5 million records were only executed once.

Exec time accuracy was 0.1s for PDI and 1ms for TOS.

Hardware Configuration:

- JVM: 1.5.0_12
- OS: Windows XP SP2
- CPU: Intel Core2 Duo T5200 @ 1,60 GHz
- RAM: 1 GB


Column	Key	Type	Nullable	Date Patte...	Length	Preci...	De...	Com...
 id	<input checked="" type="checkbox"/>	int	<input type="checkbox"/>		10			
phone	<input type="checkbox"/>	int	<input type="checkbox"/>		10			
firstname	<input type="checkbox"/>	String	<input type="checkbox"/>		50			
lastname	<input type="checkbox"/>	String	<input type="checkbox"/>		50			
addr	<input type="checkbox"/>	String	<input type="checkbox"/>		200			
code_states	<input type="checkbox"/>	String	<input type="checkbox"/>		2			
birth	<input type="checkbox"/>	Date	<input type="checkbox"/>	"yyyy-MM-...				

Figure 1: schema in TOS Job Designer

	Name	Type	Format	Position	Length	Precision
1	id	Integer			10	0
2	phone	Integer			10	0
3	firstname	String			50	
4	lastname	String			50	
5	addr	String			200	
6	code_states	String			2	
7	birth	Date	yyyy-MM-dd		10	

Figure 2: schema in PDI Spoon





	TOS 2.1.0RC1
	TOS 2.1.r4725
	PDI 2.5.0
	PDI 3.0.0M1

Figure 3: Chart callout list

Test 1: Text Input file > Text Output file

Job description

Reading x lines from a source file and writing them into a target file.

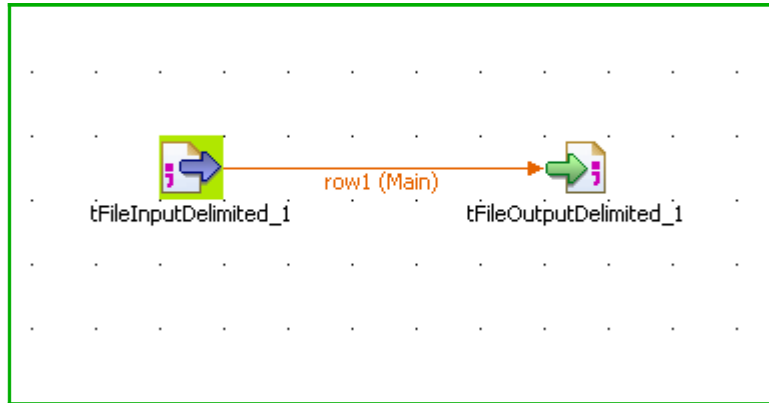


Figure 4: test 1 with TOS

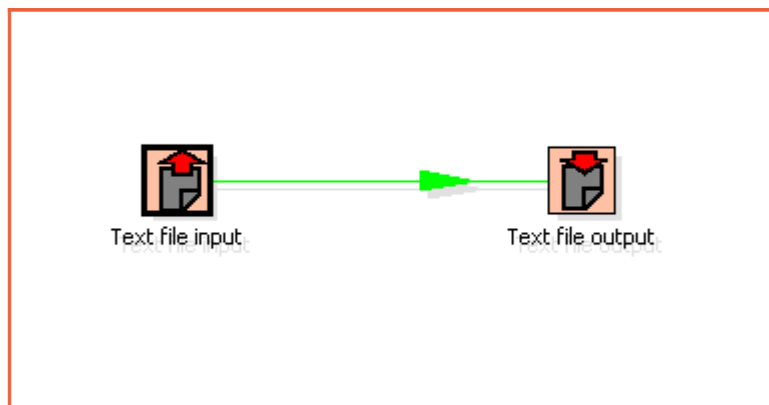
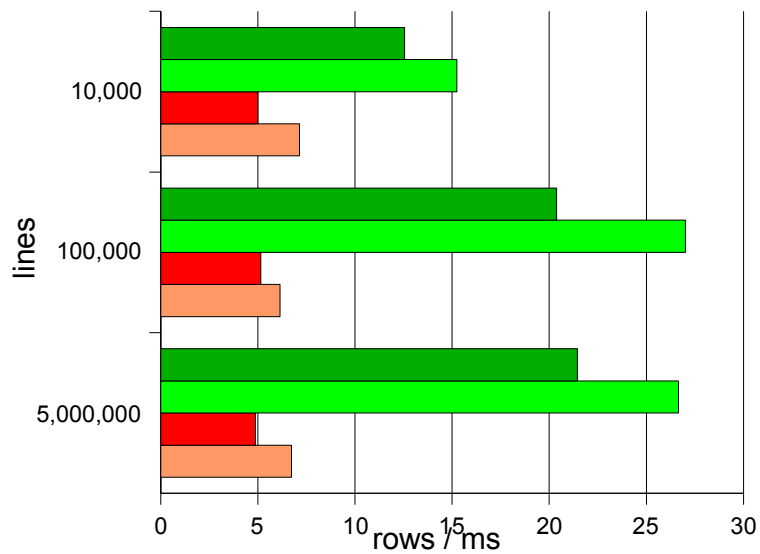


Figure 5: test 1 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	797	656	2000	1400
<i>100,000</i>	4907	3703	19400	16300
<i>5,000,000</i>	233141	187594	1025300	743400
Rows / ms				
<i>10,000</i>	12.55	15.24	5	7.14
<i>100,000</i>	20.38	27.01	5.15	6.13
<i>5,000,000</i>	21.45	26.65	4.88	6.73
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		21%	-60%	-43%
<i>100,000</i>		33%	-75%	-70%
<i>5,000,000</i>		24%	-77%	-69%

Performance chart



(the longer the stack is, the better the result)

Test 2: Text Input file > XML Output file

Job description

Reading X lines from a source file and writing them into a target file, following an XML syntax

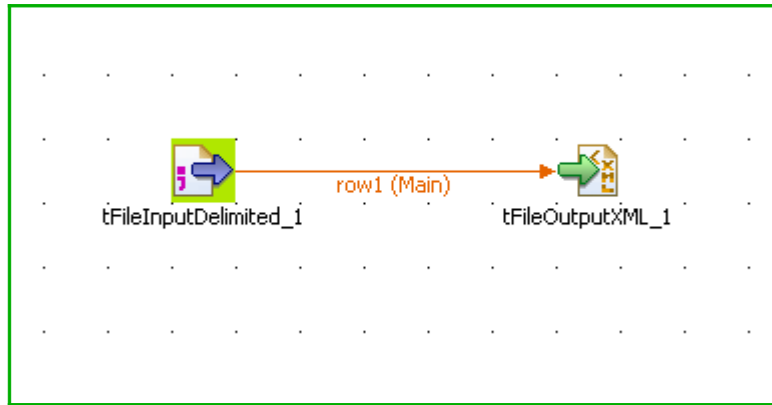


Figure 6: Test 2 with TOS

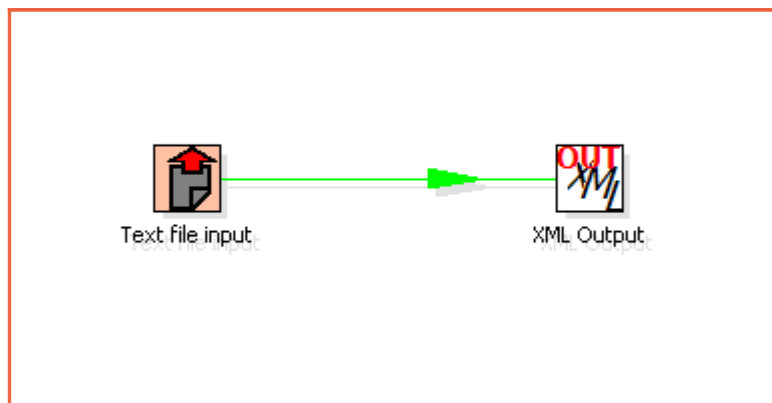
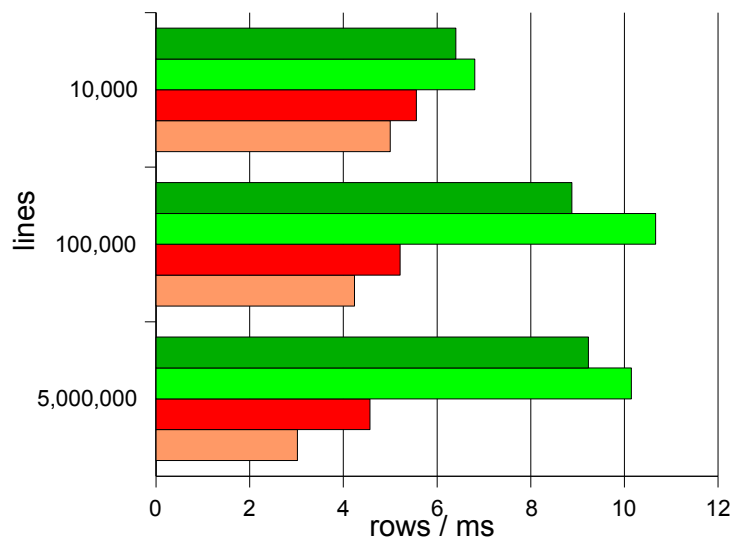


Figure 7: Test 2 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	1,562	1,469	1,800	2,000
<i>100,000</i>	11,265	9,375	19,200	23,600
<i>5,000,000</i>	541,609	492,750	1,094,600	1,656,800
Rows / ms				
<i>10,000</i>	6.4	6.81	5.56	5
<i>100,000</i>	8.88	10.67	5.21	4.24
<i>5,000,000</i>	9.23	10.15	4.57	3.02
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		6%	-13%	-22%
<i>100,000</i>		20%	-41%	-52%
<i>5,000,000</i>		10%	-51%	-67%

Performance chart



Test 3: Text Input file > Mysql Output table

Job description

Reading X lines from a source file, and writing them into a MySQL table, committing every 100 lines.

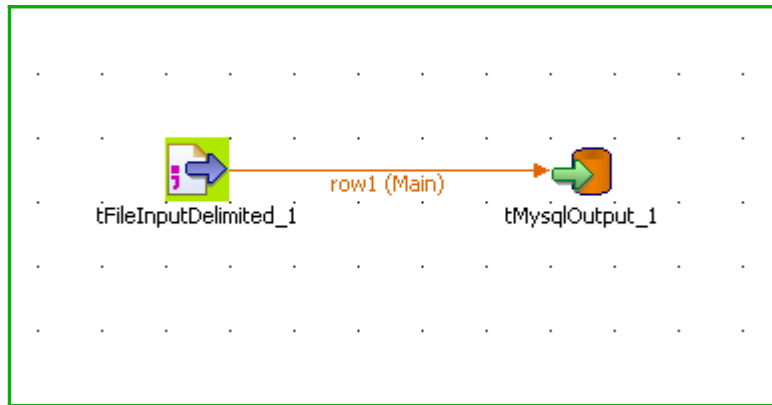


Figure 8: Test 3 with TOS

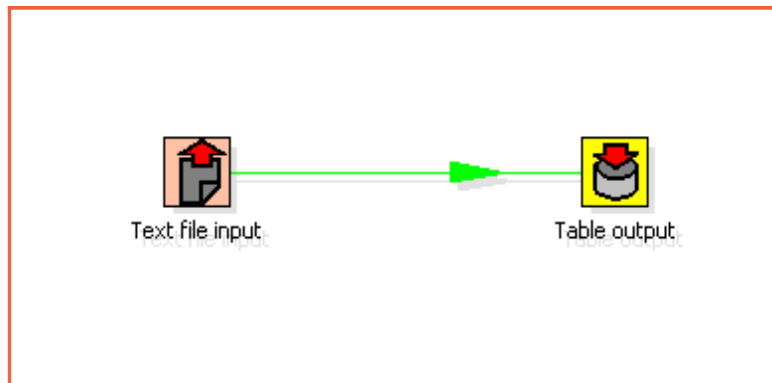
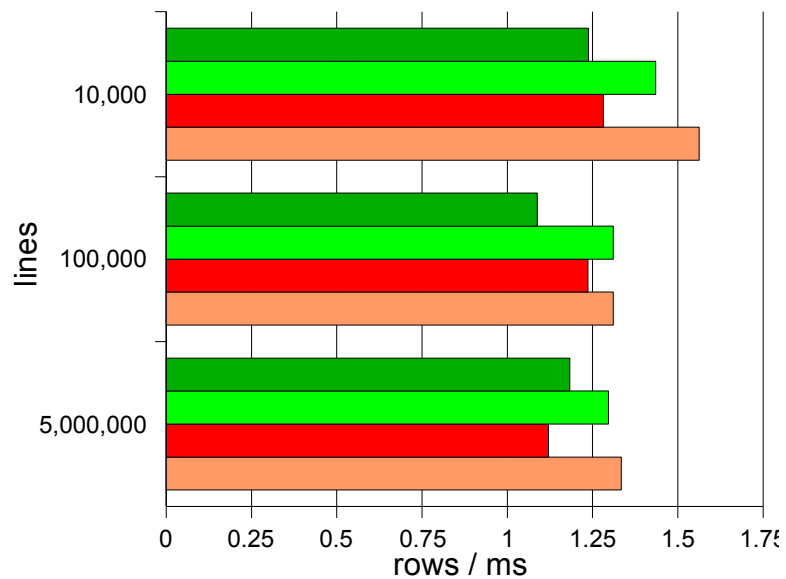


Figure 9: Test 3 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	8,078	6,968	7,800	6,400
<i>100,000</i>	91,891	76,291	80,900	76,300
<i>5,000,000</i>	4,226,125	3,854,343	4,461,500	3,746,500
Rows / ms				
<i>10,000</i>	1.24	1.44	1.28	1.56
<i>100,000</i>	1.09	1.31	1.24	1.31
<i>5,000,000</i>	1.18	1.3	1.12	1.33
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		16%	4%	26%
<i>100,000</i>		20%	14%	20%
<i>5,000,000</i>		10%	-5%	13%

Performance chart



Test 4: Text Input file > Transformation > Text Output file

Job description

Reading X lines from a source file, carrying out the following transformations:

- adding a *surrogatekey* column (sequence)
- $id = id * 7$
- $name = firstname + ' ' + lastname$
- $addr = uppercase(addr)$

Writing the transformation output into a target text file.

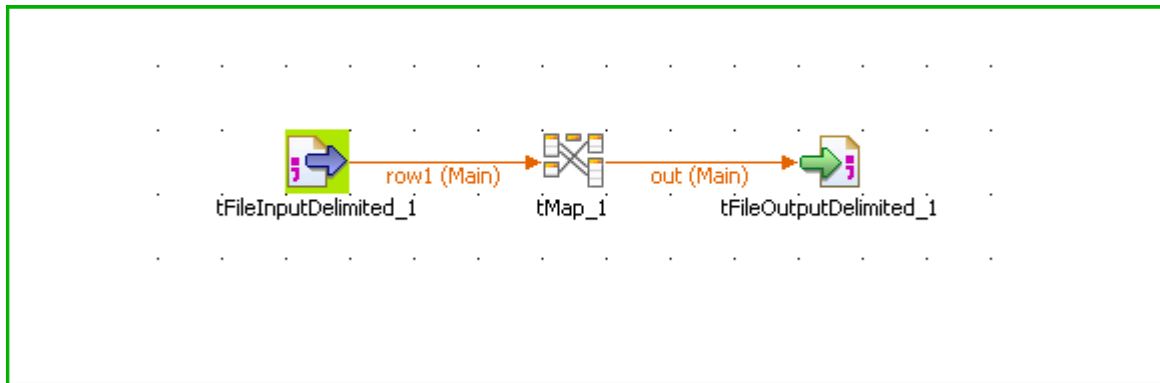


Figure 10: Test 4 with TOS

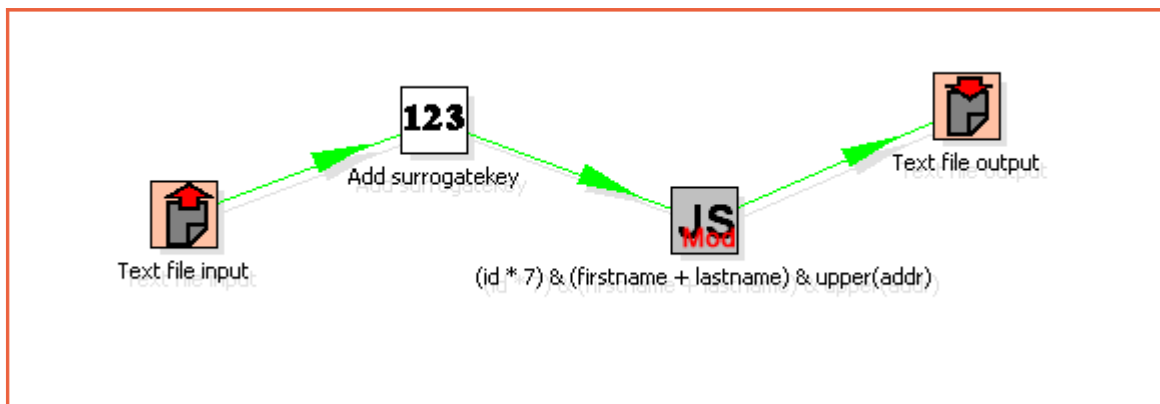
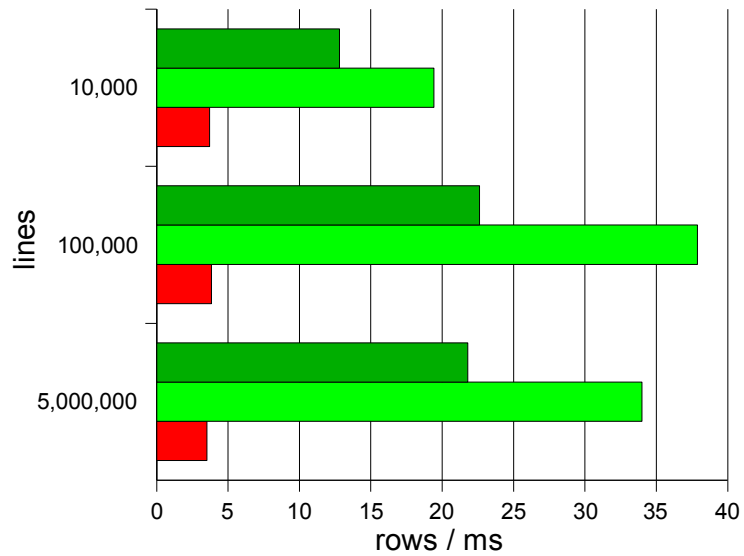


Figure 11: Test 4 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	781	515	2,700	**error**
<i>100,000</i>	4,422	2,640	26,000	**error**
<i>5,000,000</i>	229,337	147,125	1,419,100	**error**
Rows / ms				
<i>10,000</i>	12.8	19.42	3.7	**error**
<i>100,000</i>	22.61	37.88	3.85	**error**
<i>5,000,000</i>	21.8	33.98	3.52	**error**
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		52%	-71%	**error**
<i>100,000</i>		68%	-83%	**error**
<i>5,000,000</i>		56%	-84%	**error**

Performance chart



Test 5: Test 4 + Lookup

Job description

Reading X lines from a source file, carrying out the transformations as specified in Test 4, looking up to a MySQL table, for a State name using *code_state* column. Then writing the transformation output into a target file.

Notes

Lookup table is cached in PDI.

Lookup table size = 1296 rows.

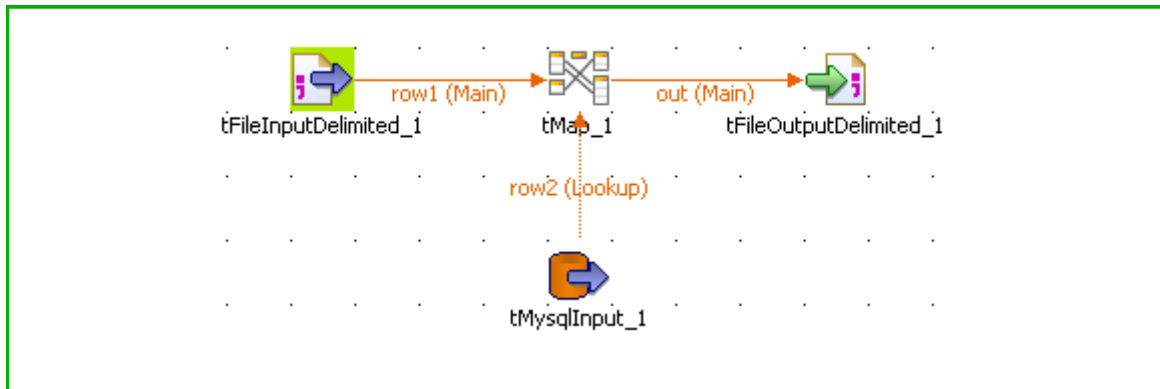


Figure 12: Test 5 with TOS

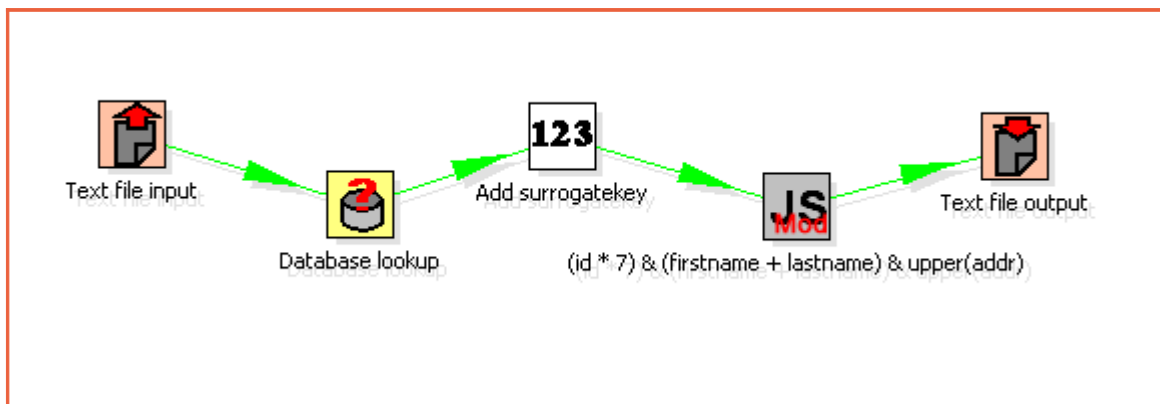
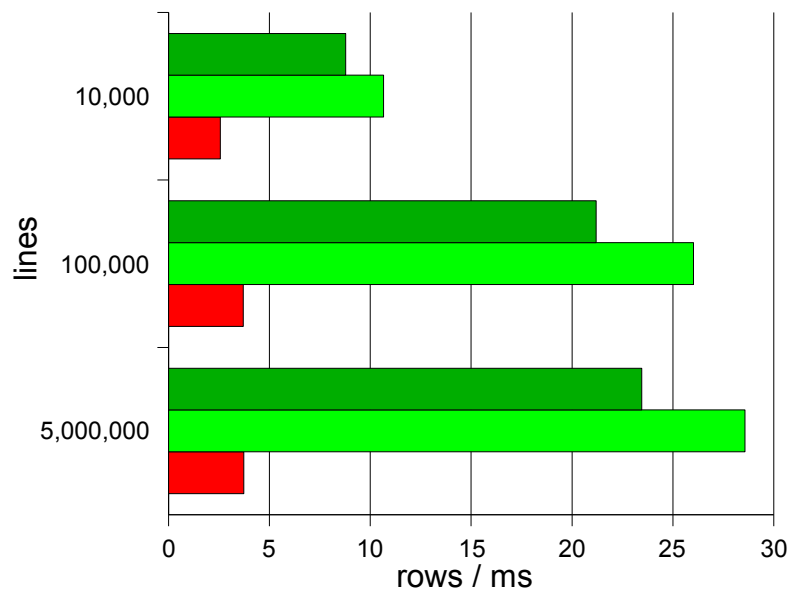


Figure 13: Test 5 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	1,140	938	3,900	**error**
<i>100,000</i>	4,719	3,843	27,000	**error**
<i>5,000,000</i>	213,187	174,937	1,341,500	**error**
Rows / ms				
<i>10,000</i>	8.77	10.66	2.56	**error**
<i>100,000</i>	21.19	26.02	3.7	**error**
<i>5,000,000</i>	23.45	28.58	3.73	**error**
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		22%	-71%	**error**
<i>100,000</i>		23%	-83%	**error**
<i>5,000,000</i>		22%	-84%	**error**

Performance chart



Test 6: Test 5 + output filter

Job description

Reading X lines from a source file, carrying out the transformations as specified in Test 5, filtering the output (*code_state* matching 'FR'), writing the filtered output into a first target text file. Writing the main output flow into a second target text file.

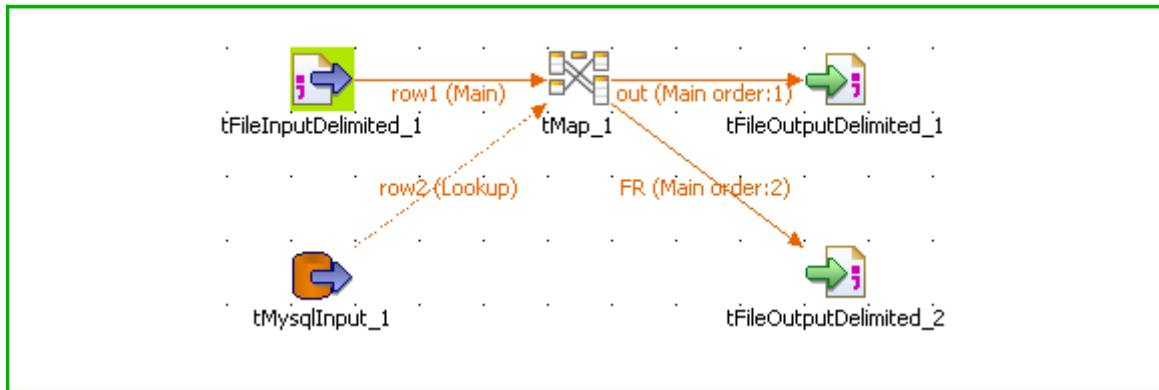


Figure 14: Test 6 with TOS

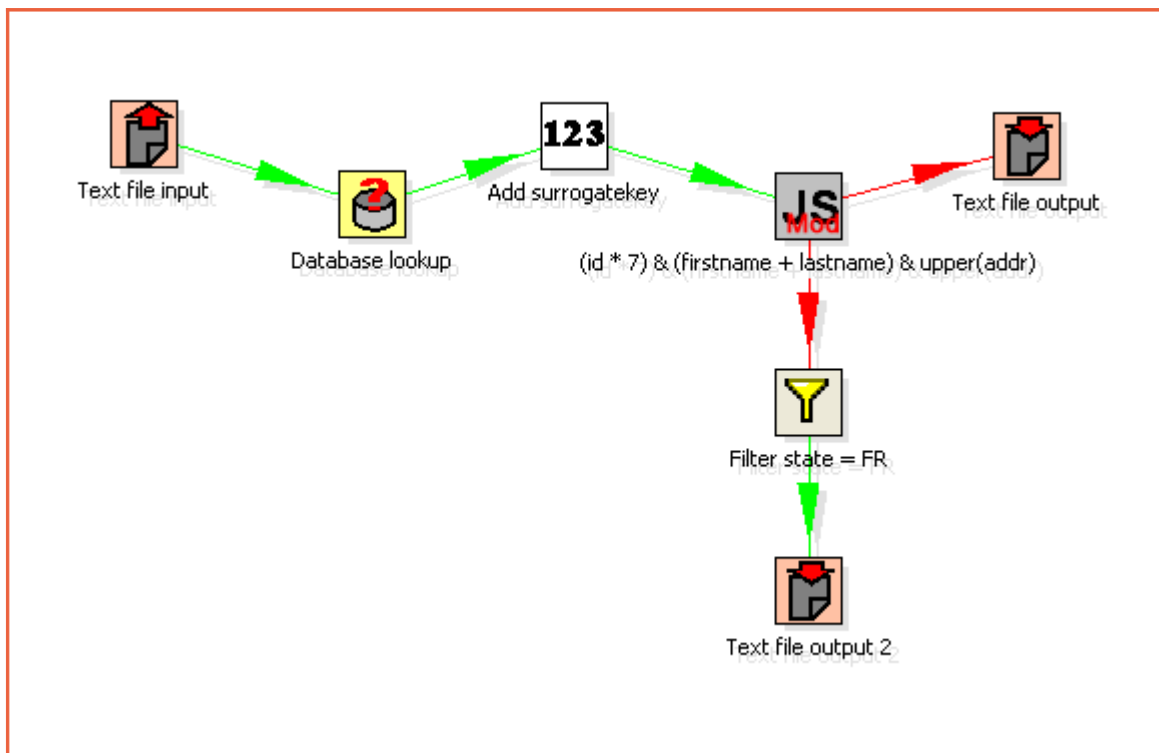
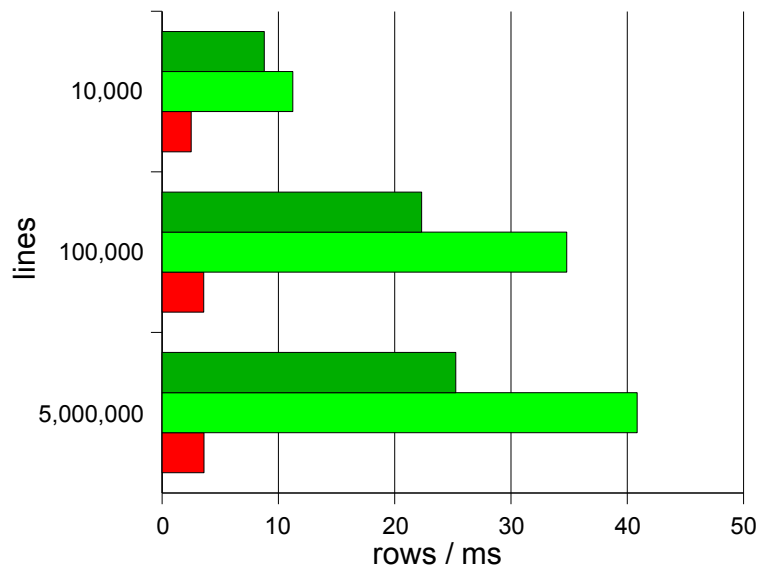


Figure 15: Test 6 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	1,140	890	4,000	**error**
<i>100,000</i>	4,484	2,875	28,000	**error**
<i>5,000,000</i>	198,000	122,391	1,388,400	**error**
Rows / ms				
<i>10,000</i>	8.77	11.24	2.5	**error**
<i>100,000</i>	22.3	34.78	3.57	**error**
<i>5,000,000</i>	25.25	40.85	3.6	**error**
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		28%	-72%	**error**
<i>100,000</i>		56%	-84%	**error**
<i>5,000,000</i>		62%	-86%	**error**

Performance chart



Test 7: Test 6 + aggregation

Job description

Reading X lines from a source file, carrying out the transformations as specified in Test 6. The main output flow is aggregated on the *code_state* column, and SUM, MAX, MIN, AVG functions are applied on column *id*.

Notes

PDI sorts and writes every 400 000 lines in a file, in order to reduce the memory use. The rows should be sorted before aggregation in PDI.

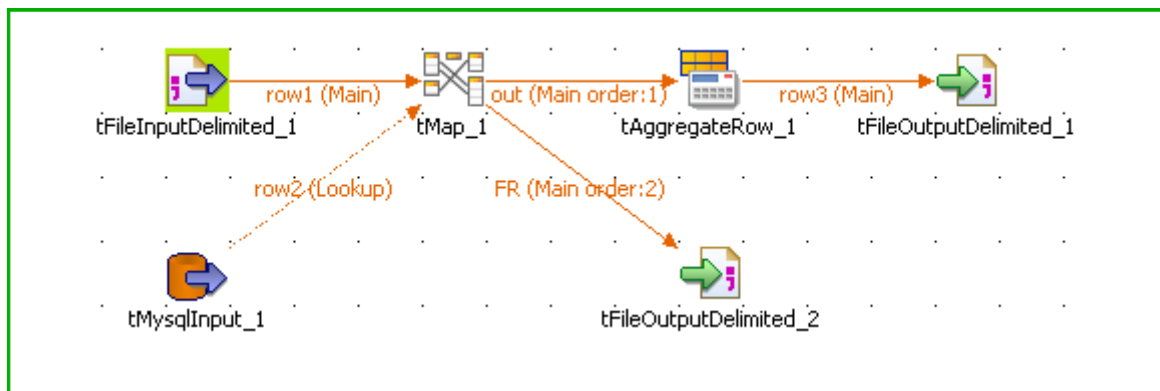


Figure 16: Test 7 with TOS

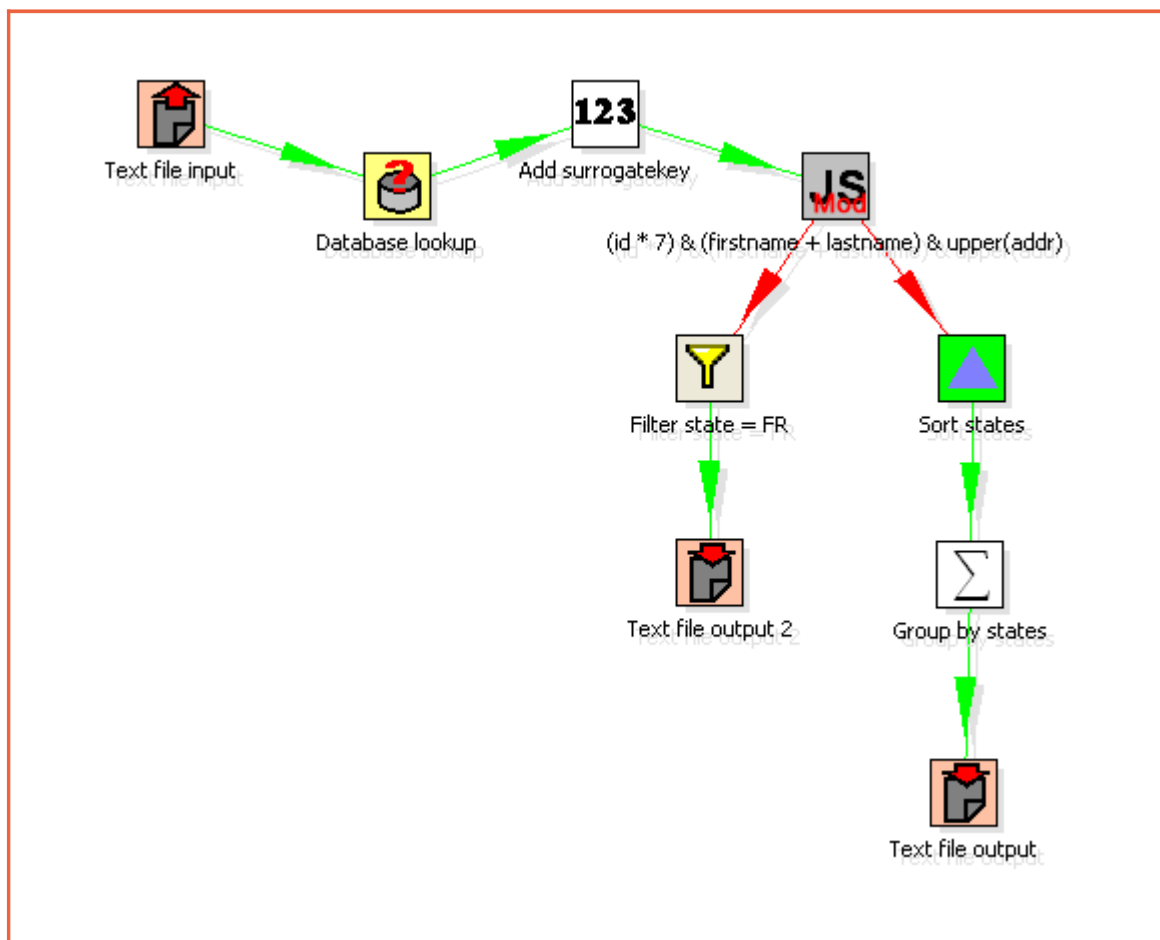
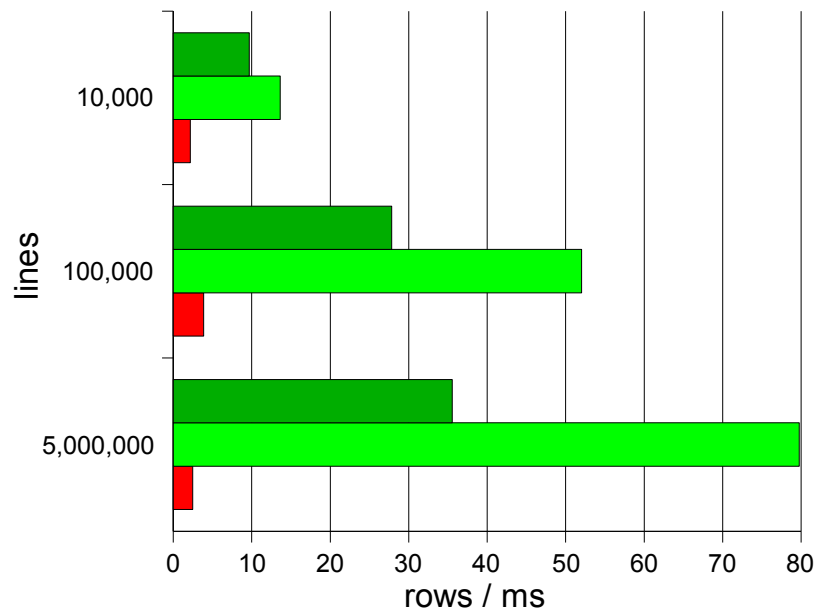


Figure 17: Test 7 with PDI

Test Results

Nr of lines	TOS 2.1.0RC1	TOS r4725	PDI 2.5	PDI 3.0M1
Exec time (ms)				
<i>10,000</i>	1,032	734	4,600	**error**
<i>100,000</i>	3,594	1,922	25,800	**error**
<i>5,000,000</i>	140,688	62,672	2,015,000	**error**
Rows / ms				
<i>10,000</i>	9.69	13.62	2.17	**error**
<i>100,000</i>	27.82	52.03	3.88	**error**
<i>5,000,000</i>	35.54	79.78	2.48	**error**
Ratio of Nr of rows processed/ms (against TOS 2.1.0RC1 results)				
<i>10,000</i>		41%	-78%	**error**
<i>100,000</i>		87%	-86%	**error**
<i>5,000,000</i>		124%	-93%	**error**

Performance chart



Appendix 1: Transformation step/component

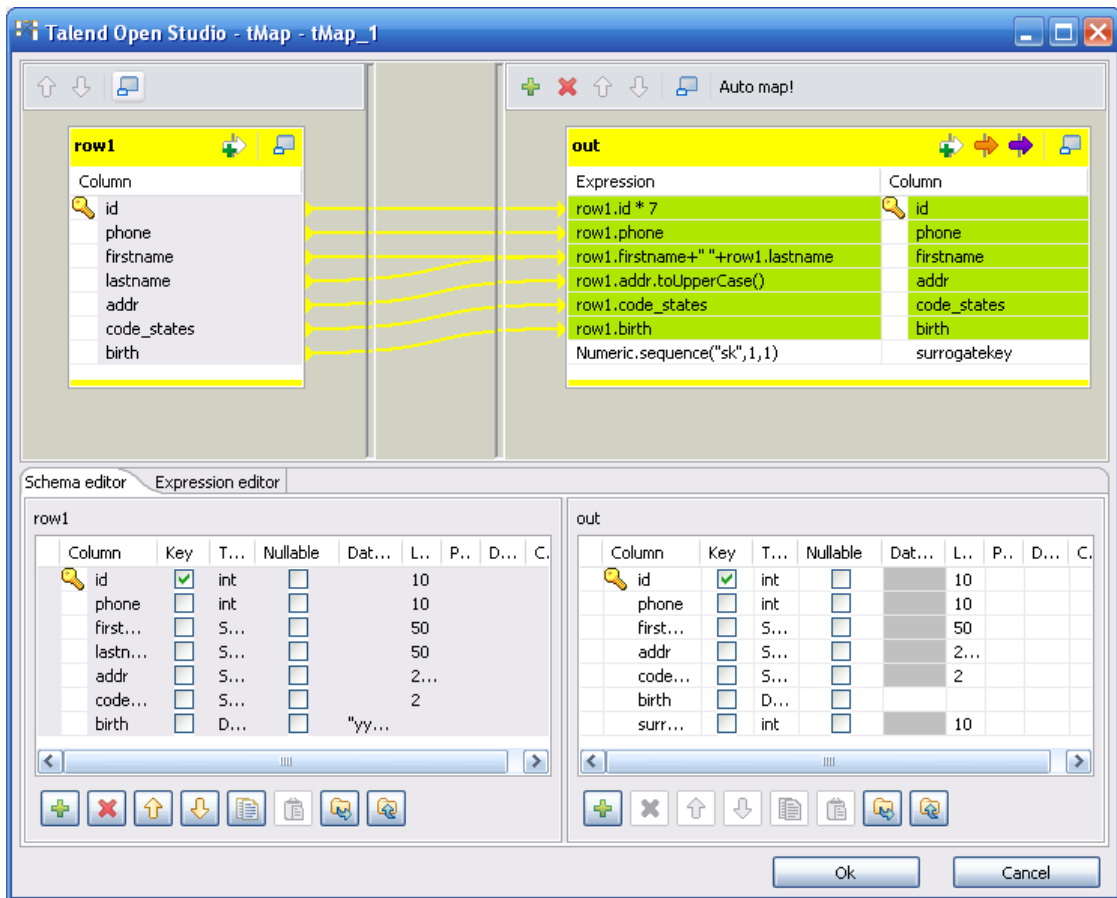


Figure 18: Test 4 tMap component details (TOS)

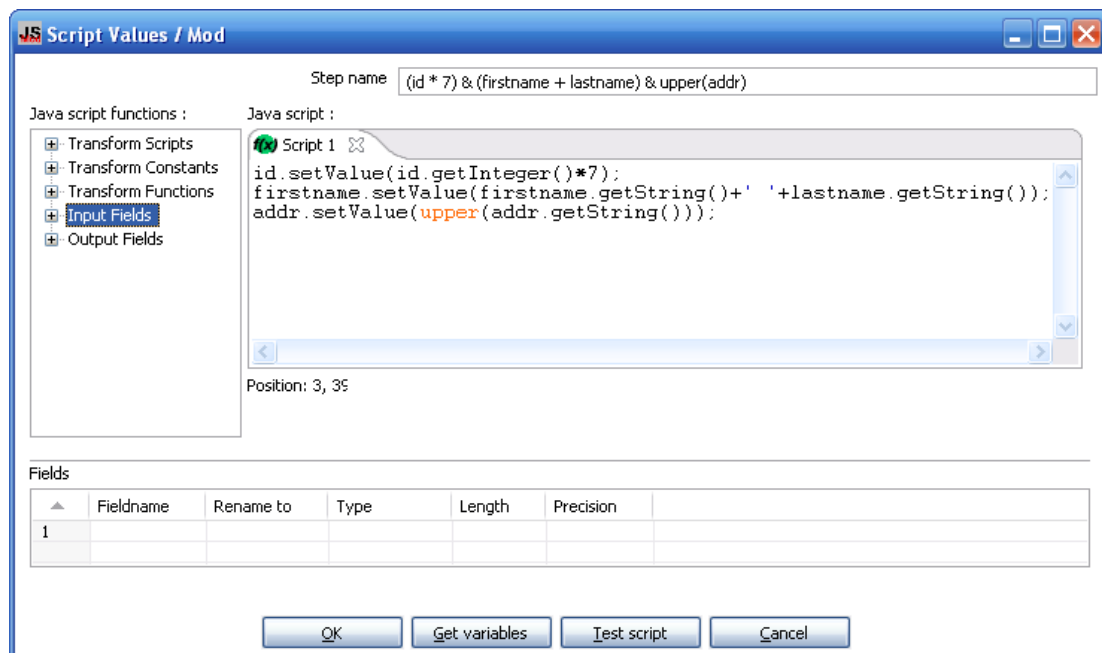


Figure 19: Test 4 JavaScript transformation step details (PDI)