

To: MTB Distribution
From: Roger Lackey
Date: September 24, 1981
Subject: MR9.0 MRDS Performance Results

PURPOSE

The purpose of this MTB is to show the performance differences between MR6.5 MRDS and the MR9.0 MRDS.

I. SUMMARY RESULTS

The following table shows the relationship between MRDS 6.5 and MRDS 9.0 performance.

FUNCTION	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
store	2.70	0.69	74.44	3.91
retrieve	274.92	103.92	62.20	2.65
modify	271.97	135.51	50.18	2.01
delete	127.52	80.10	37.19	1.59
open	23.95	8.79	63.30	2.73
close	7.80	2.64	66.15	2.96
Overall: Average			62.12	2.64 *

* Calculated by averaging the times faster for the six test categories.

Multics Project internal working documentation. Not to be reproduced outside the Multics Project.

II. SUMMARY RESULTS between MR8.0 and MR9.0 MRDS

The following table shows the percentage gain and times faster statistics for both MR8.0 and MR9.0 MRDS as measured against MR6.5 MRDS.

	MR8.0 *	TIMES %GAIN FASTER	MR9.0 %GAIN	TIMES FASTER
store	6.91	1.07	74.44	3.91
retrieve	74.86	3.98	62.20	2.65
modify	8.47	1.09	50.18	2.01
delete	11.41	1.13	37.19	1.59
open	14.46	1.70	63.30	2.73
close	-2.39	0.99	66.15	2.96
Average		1.66		2.64

The improvement between MR8.0 and MR9.0 MRDS is 1.59 times faster.

The decrease in the retrieval performance between MR8.0 and MR9.0 is attributed to the fact that these test were for a one relation database and MR9.0 has added considerable optimization for two or more relations in a selection expression.

* As published in MTB 450 for MR8.0 MRDS.

III. TOTALS FOR EACH FUNCTION

FUNCTION	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
store:				
fixed bin	1.214	0.389	67.957	3.12
char	1.485	0.300	79.798	4.90
Total	2.70	0.69	74.44	3.91
retrieve:				
er fixed bin	83.86	17.98	78.56	4.66
er char	53.68	17.89	66.67	3.00
er.all	137.53	35.87	73.92	3.83
eu fixed bin	83.81	34.12	59.29	2.46
eu char	53.59	33.93	36.67	1.58
eu.all	137.39	68.05	50.47	2.02
Total	274.92	103.92	62.20	2.65
modify:				
data	80.22	40.44	49.44	1.98
index	94.79	41.91	55.79	2.26
data and index	96.96	53.16	45.17	1.82
Total	271.97	135.51	50.18	2.01
delete:				
fixed bin	77.11	39.47	48.82	1.96
char	50.41	40.63	19.40	1.24
Total	127.52	80.10	37.19	1.59
open:				
er	11.23	4.11	63.37	2.73
eu	12.72	4.68	63.18	2.72
Total	23.95	8.79	63.30	2.73
close:				
er	3.88	1.33	65.77	2.92
eu	3.92	1.31	66.55	2.99
Total	7.80	2.64	66.15	2.96

IV. TEST DESCRIPTIONS

All tests were performed against a one relation database with character data type and then against the same database but with a fixed binary data type for each attribute.

The character database described as follows:

domain: common char (20);

attribute:

k001 common ,
k002 common ,
d001 common ,
x001 common ;

relation:

r001 (k001* k002* d001 x001);

index:

r001 (x001);

and the fixed binary database described as follows.

domain: common fixed_bin;

attribute:

k001 common ,
k002 common ,
d001 common ,
x001 common ;

relation:

r001 (k001* k002* d001 x001);

index:

r001 (x001);

The selection expressions used in retrieve, modify and delete tests span only one relation. The tests consist of three sets: the first selects the first 10 tuples (keys 1 - 10), the middle which selects the middle 10 tuples (keys 496 - 505), and the last 10 tuples (keys 991 - 1000).

The data was acquired by a program that given a MRDS function (retrieve, modify delete, etc), a database index and where needed a selection expression, executes the test reading the virtual clock before and after the call to MRDS. It make the call to MRDS in such a way that no data conversion is encountered.

The results are expressed as a percentage improvement of new MRDS to old MRDS as calculated by the formula:

$$\text{PERCENT GAIN} = ((\text{old_time} - \text{new_time}) / \text{old_time}) * 100$$

$$\text{TIMES FASTER} = 100 / (100 - \text{PERCENT GAIN})$$

The modify test modified a data attribute in the first test, an indexed attribute in the second test and both a data and an index attribute in the third test. A problem with MR 6.5 MRDS prevented the timing of the modifies for the character data type.

The store tests consisted of storing 1000 tuples in each of the fixed binary and character databases.

The open and close tests were run against the database with both exclusive retrieval and exclusive update opening modes.

APPENDIX A

The following is the detailed data of each test.

Store 1000 tuples

	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
store fixed bin	1.214	0.389	67.957	3.121
store char	1.485	0.300	79.798	4.950
store TOTAL	2.70	0.69	74.44	3.91

retrieve opened exclusive retrieve fixed bin data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	22.066	7.176	67.479	3.075
2	4.197	1.193	71.575	3.518
3	4.193	0.374	91.080	11.211
4	3.598	0.343	90.467	10.490
5	3.666	1.183	67.730	3.099
6	3.690	0.347	90.596	10.634
7	4.206	1.193	71.636	3.526
8	4.301	0.341	92.072	12.613
9	3.652	0.339	90.717	10.773
10	3.698	1.187	67.902	3.115
11	3.676	0.351	90.452	10.473
12	5.020	1.431	71.494	3.508
13	5.025	0.343	93.174	14.650
14	4.279	0.345	91.937	12.403
15	4.263	1.413	66.854	3.017
16	4.327	0.419	90.317	10.327
fixed bin TOTAL	83.86	17.98	8.56	4.66

retrieve open exclusive retrieve char data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	22.446	7.238	67.754	3.101
2	3.703	1.173	68.323	3.157
3	0.530	0.351	33.774	1.510
4	3.110	0.334	89.260	9.311
5	3.098	1.147	62.976	2.701
6	3.104	0.353	88.628	8.793
7	3.664	1.176	67.904	3.116
8	0.522	0.377	27.778	1.385
9	0.495	0.354	28.485	1.398
10	3.098	1.184	61.782	2.617
11	0.537	0.352	34.451	1.526
12	4.171	1.385	66.795	3.012
13	0.548	0.354	35.401	1.548
14	0.549	0.358	34.791	1.534
15	3.482	1.384	60.253	2.516
16	0.618	0.371	39.968	1.666
char TOTAL	53.68	17.89	66.67	3.00
retrieve.er TOTAL	137.53	35.87	73.92	3.83

retrieve open exclusive update fixed bin data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	22.146	9.360	57.735	2.366
2	4.276	3.534	17.353	1.210
3	4.291	0.365	91.494	11.756
4	3.624	0.338	90.673	10.722
5	3.632	3.512	3.304	1.034
6	3.647	0.342	90.622	10.664
7	4.229	3.506	17.096	1.206
8	4.209	0.375	91.091	11.224
9	3.626	0.352	90.292	10.301
10	3.547	3.556	-0.254	0.997
11	3.632	0.386	89.372	9.409
12	5.024	3.705	26.254	1.356
13	5.043	0.351	93.040	14.368
14	4.258	0.360	91.545	11.828
15	4.294	3.706	13.694	1.159
16	4.329	0.370	91.453	11.700
TOTAL fixed bin	83.81	34.12	59.29	2.46

retrieve open exclusive update char data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	22.541	9.296	58.760	2.425
2	3.623	3.483	3.864	1.040
3	0.514	0.337	34.436	1.525
4	3.087	0.346	88.792	8.922
5	3.066	3.591	-17.123	0.854
6	3.125	0.357	88.576	8.754
7	3.698	3.535	4.408	1.046
8	0.533	0.336	36.961	1.586
9	0.492	0.338	31.301	1.456
10	3.047	3.522	-15.589	0.865
11	0.522	0.368	29.502	1.418
12	4.168	3.699	11.252	1.127
13	0.555	0.352	36.577	1.577
14	0.551	0.349	36.661	1.579
15	3.440	3.653	-6.192	0.942
16	0.623	0.372	40.289	1.675
TOTAL char	53.59	33.93	36.67	1.58
retrieve.eu TOTAL	137.39	68.05	50.47	2.02
retrieve TOTAL	274.92	103.92	62.20	2.65

modify data fixed bin data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	19.157	12.919	32.563	1.483
2	4.091	3.741	8.555	1.094
3	4.116	0.514	87.512	8.008
4	3.454	0.494	85.698	6.992
5	3.492	3.747	-7.302	0.932
6	3.633	0.499	86.265	7.281
7	4.310	3.881	9.954	1.111
8	4.225	0.529	87.479	7.987
9	3.526	0.577	83.636	6.111
10	3.572	3.768	-5.487	0.948
11	3.602	0.587	83.703	6.136
12	5.045	3.916	22.379	1.288
13	5.172	0.506	90.217	10.221
14	4.229	0.556	86.853	7.606
15	4.253	3.808	10.463	1.117
16	4.346	0.518	88.081	8.390
modify data TOTAL	80.22	40.44	49.44	1.98

modify index fixed bin data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	30.888	22.767	26.292	1.357
2	4.398	0.621	85.880	7.082
3	4.299	0.512	88.090	8.396
4	3.812	0.562	85.257	6.783
5	3.645	3.835	-5.213	0.950
6	3.801	0.572	84.951	6.645
7	4.349	0.592	86.388	7.346
8	4.389	0.610	86.102	7.195
9	3.631	0.614	83.090	5.914
10	3.614	3.910	-8.190	0.924
11	3.778	0.682	81.948	5.540
12	5.391	0.612	88.648	8.809
13	5.227	0.605	88.425	8.640
14	4.657	0.621	86.665	7.499
15	4.522	4.104	9.244	1.102
16	4.390	0.688	84.328	6.381
modify index TOTAL	94.79	41.91	55.79	2.26

modify data_index fixed bin data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	32.929	23.368	29.035	1.409
2	4.504	3.921	12.944	1.149
3	4.528	0.594	86.882	7.623
4	3.534	0.583	83.503	6.062
5	3.597	3.893	-8.229	0.924
6	3.795	0.611	83.900	6.211
7	4.354	3.934	9.646	1.107
8	4.255	0.661	84.465	6.437
9	3.671	0.625	82.975	5.874
10	3.640	3.911	-7.445	0.931
11	3.735	0.665	82.195	5.617
12	5.297	4.299	18.841	1.232
13	5.333	0.601	88.731	8.874
14	4.653	0.631	86.439	7.374
15	4.637	4.216	9.079	1.100
16	4.501	0.651	85.537	6.914
modify di TOTAL	96.96	53.16	45.17	1.82
modify TOTAL	271.97	135.51	50.18	2.01

delete fixed bin data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	20.189	15.024	25.583	1.344
2	4.015	3.401	15.293	1.181
3	3.991	0.399	90.003	10.003
4	3.339	0.368	88.979	9.073
5	3.431	3.372	1.720	1.017
6	3.386	0.387	88.571	8.749
7	3.936	3.385	13.999	1.163
8	3.898	0.369	90.534	10.564
9	3.317	0.369	88.875	8.989
10	3.347	3.384	-1.105	0.989
11	3.260	0.384	88.221	8.490
12	4.544	3.681	18.992	1.234
13	4.598	0.376	91.823	12.229
14	3.949	0.394	90.023	10.023
15	3.926	3.744	4.636	1.049
16	3.986	0.435	89.087	9.163
TOTAL delete fixed	77.11	39.47	48.82	1.96

delete character data

SEL-EXP NUMBER	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
1	19.666	15.795	19.684	1.245
2	3.308	3.458	-4.534	0.957
3	2.601	0.399	84.660	6.519
4	2.714	0.390	85.630	6.959
5	2.789	3.407	-22.158	0.819
6	2.765	0.396	85.678	6.982
7	3.415	3.443	-0.820	0.992
8	0.283	0.384	-35.689	0.737
9	0.266	0.374	-40.602	0.711
10	2.717	3.457	-27.236	0.786
11	0.297	0.386	-29.966	0.769
12	3.843	3.739	2.706	1.028
13	1.842	0.410	77.742	4.493
14	0.326	0.406	-24.540	0.803
15	3.186	3.750	-17.702	0.850
16	0.390	0.433	-11.026	0.901
delete TOTAL char	50.41	40.63	19.40	1.24
delete TOTAL	127.52	80.10	37.19	1.59

open exclusive retrieval (6 successive openings)

	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
	1.892	0.700	63.002	2.703
	1.893	0.686	63.761	2.759
	1.830	0.678	62.951	2.699
	1.898	0.694	63.435	2.735
	1.860	0.680	63.441	2.735
	1.854	0.674	63.646	2.751
open.er TOTAL	11.23	4.11	63.37	2.73

open exclusive update (6 successive openings)

	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
	2.039	0.851	58.264	2.396
	2.088	0.743	64.416	2.810
	2.191	0.825	62.346	2.656
	2.090	0.768	63.254	2.721
	2.060	0.734	64.369	2.807
	2.248	0.761	66.148	2.954
open.eu TOTAL	12.72	4.68	63.18	2.72
open TOTAL	23.95	8.79	63.30	2.73

close exclusive retrieval (of 6 successive openings)

	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
	0.656	0.212	67.683	3.094
	0.668	0.224	66.467	2.982
	0.629	0.220	65.024	2.859
	0.631	0.243	61.490	2.597
	0.660	0.215	67.424	3.070
	0.638	0.215	66.301	2.967
close.er TOTAL	3.88	1.33	65.77	2.92

close exclusive update (of 6 successive openings)

	MR6.5 TIME	MR9.0 TIME	PERCENT GAIN	TIMES FASTER
	0.641	0.218	65.991	2.940
	0.644	0.209	67.547	3.081
	0.685	0.217	68.321	3.157
	0.663	0.237	64.253	2.797
	0.630	0.216	65.714	2.917
	0.650	0.212	67.385	3.066
close.eu TOTAL	3.92	1.31	66.55	2.99
close TOTAL	7.80	2.64	66.15	2.96

APPENDIX B

Blanks have been removed from the quoted strings for readability.

SEL-EXP FOR RETRIEVE AND DELETE
NUMBER

- 1 "-range (r r001) -select r"
- 2 "-range (r r001) -select r -where r.d001 <= ""10""
- 3 "-range (r r001) -select r -where r.x001 <= ""10""
- 4 "-range (r r001) -select r -where r.k001 <= ""10""
- 5 "-range (r r001) -select r -where r.k002 <= ""10""
- 6 "-range (r r001) -select r -where ((r.k001 <= ""10"") &
 (r.k002 <= ""10""))"
- 7 "-range (r r001) -select r -where r.d001 > ""990""
- 8 "-range (r r001) -select r -where r.x001 > ""990""
- 9 "-range (r r001) -select r -where r.k001 > ""990""
- 10 "-range (r r001) -select r -where r.k002 > ""990""
- 11 "-range (r r001) -select r -where ((r.k001 > ""990"") &
 (r.k002 > ""990""))"
- 12 "-range (r r001) -select r -where ((r.d001 > ""495"") &
 (r.d001 < ""506""))"
- 13 "-range (r r001) -select r -where ((r.x001 > ""495"") &
 (r.x001 < ""506""))"
- 14 "-range (r r001) -select r -where ((r.k001 > ""495"") &
 (r.k001 < ""506""))"
- 15 "-range (r r001) -select r -where ((r.k002 > ""495"") &
 (r.k002 < ""506""))"
- 16 "-range (r r001) -select r -where (((r.k001 > ""495"") &
 (r.k001 < ""506"")) & ((r.k002 > ""495"") &
 (r.k002 < ""506"")))"

Blanks have been removed from the quoted strings for readability.

SEL-EXP FOR MODIFY DATA

```
1  "-range (r r001) -select r.d001"
2  "-range (r r001) -select r.d001 -where r.x001 <= "10""
3  "-range (r r001) -select r.d001 -where r.x001 <= "10""
4  "-range (r r001) -select r.d001 -where r.k001 <= "10""
5  "-range (r r001) -select r.d001 -where r.k002 <= "10""
6  "-range (r r001) -select r.d001 -where ((r.k001 <= "10") &
   (r.k002 <= "10"))"
7  "-range (r r001) -select r.d001 -where r.x001 > "990""
8  "-range (r r001) -select r.d001 -where r.x001 > "990""
9  "-range (r r001) -select r.d001 -where r.k001 > "990""
10 "-range (r r001) -select r.d001 -where r.k002 > "990""
11 "-range (r r001) -select r.d001 -where ((r.k001 > "990") &
   (r.k002 > "990"))"
12 "-range (r r001) -select r.d001 -where ((r.x001 > "495") &
   (r.x001 < "506"))"
13 "-range (r r001) -select r.d001 -where ((r.x001 > "495") &
   (r.x001 < "506"))"
14 "-range (r r001) -select r.d001 -where ((r.k001 > "495") &
   (r.k001 < "506"))"
15 "-range (r r001) -select r.d001 -where ((r.k002 > "495") &
   (r.k002 < "506"))"
16 "-range (r r001) -select r.d001 -where (((r.k001 > "495") &
   (r.k001 < "506")) & ((r.k002 > "495") &
   (r.k002 < "506")))"
```

Blanks have been removed from the quoted strings for readability.

SEL-EXP FOR MODIFY INDEX

```

1  "-range (r r001) -select r.x001"
2  "-range (r r001) -select r.x001 -where r.x001 <= "10""
3  "-range (r r001) -select r.x001 -where r.x001 <= "10""
4  "-range (r r001) -select r.x001 -where r.k001 <= "10""
5  "-range (r r001) -select r.x001 -where r.k002 <= "10""
6  "-range (r r001) -select r.x001 -where ((r.k001 <= "10") &
   (r.k002 <= "10"))"
7  "-range (r r001) -select r.x001 -where r.x001 > "990""
8  "-range (r r001) -select r.x001 -where r.x001 > "990""
9  "-range (r r001) -select r.x001 -where r.k001 > "990""
10 "-range (r r001) -select r.x001 -where r.k002 > "990""
11 "-range (r r001) -select r.x001 -where ((r.k001 > "990") &
   (r.k002 > "990"))"
12 "-range (r r001) -select r.x001 -where ((r.x001 > "495") &
   (r.x001 < "506"))"
13 "-range (r r001) -select r.x001 -where ((r.x001 > "495") &
   (r.x001 < "506"))"
14 "-range (r r001) -select r.x001 -where ((r.k001 > "495") &
   (r.k001 < "506"))"
15 "-range (r r001) -select r.x001 -where ((r.k002 > "495") &
   (r.k002 < "506"))"
16 "-range (r r001) -select r.x001 -where ((r.k001 > "495") &
   (r.k001 < "506")) & ((r.k002 > "495") &
   (r.k002 < "506"))"

```


Blanks have been removed from the quoted strings for readability.

SEL-EXP FOR MODIFY DATA AND INDEX

```

1  "-range (r r001) -select r.d001 r.x001"
2  "-range (r r001) -select r.d001 r.x001 -where r.d001 <= ""10""
3  "-range (r r001) -select r.d001 r.x001 -where r.x001 <= ""10""
4  "-range (r r001) -select r.d001 r.x001 -where r.k001 <= ""10""
5  "-range (r r001) -select r.d001 r.x001 -where r.k002 <= ""10""
6  "-range (r r001) -select r.d001 r.x001 -where ((r.k001 <= ""10"" ) &
   (r.k002 <= ""10""))"
7  "-range (r r001) -select r.d001 r.x001 -where r.d001 > ""990""
8  "-range (r r001) -select r.d001 r.x001 -where r.x001 > ""990""
9  "-range (r r001) -select r.d001 r.x001 -where r.k001 > ""990""
10 "-range (r r001) -select r.d001 r.x001 -where r.k002 > ""990""
11 "-range (r r001) -select r.d001 r.x001 -where ((r.k001 > ""990"" ) &
   (r.k002 > ""990""))"
12 "-range (r r001) -select r.d001 r.x001 -where ((r.d001 > ""495"" ) &
   (r.d001 < ""506""))"
13 "-range (r r001) -select r.d001 r.x001 -where ((r.x001 > ""495"" ) &
   (r.x001 < ""506""))"
14 "-range (r r001) -select r.d001 r.x001 -where ((r.k001 > ""495"" ) &
   (r.k001 < ""506""))"
15 "-range (r r001) -select r.d001 r.x001 -where ((r.k002 > ""495"" ) &
   (r.k002 < ""506""))"
16 "-range (r r001) -select r.d001 r.x001 -where (((r.k001 > ""495"" ) &
   (r.k001 < ""506"")) & ((r.k002 > ""495"" ) &
   (r.k002 < ""506"")))"

```

APPENDIX C

EFFECTS OF ATTRIBUTE LEVEL CONTROL

The purpose of this test was to determine if the implementation of attribute level control had a significant effect on the the performance of MRDS. As shown by the following table the impact was negligible.

	UNSECURED DATABASE TIME	SECURED DATABASE TIME	PERCENT CHANGE
	67.45	69.45	-2.9
	15.26	15.48	-1.4
	27.88	26.93	+3.5
Total	110.59	111.86	0.8

The time were the timing results from the LINUS regression tests with the database being opened through a submodel.

The same test was also run against a MR6.5 version of the database using the same submodel with the following results:

	MR6.5	UNSECURED MR9.0	PERCENT GAIN	TIMES FASTER
	96.64	67.45	30.20	1.43
	29.23	15.26	47.79	1.92
	45.72	27.88	39.02	1.64
Total	171.59	110.59	35.55	1.55