

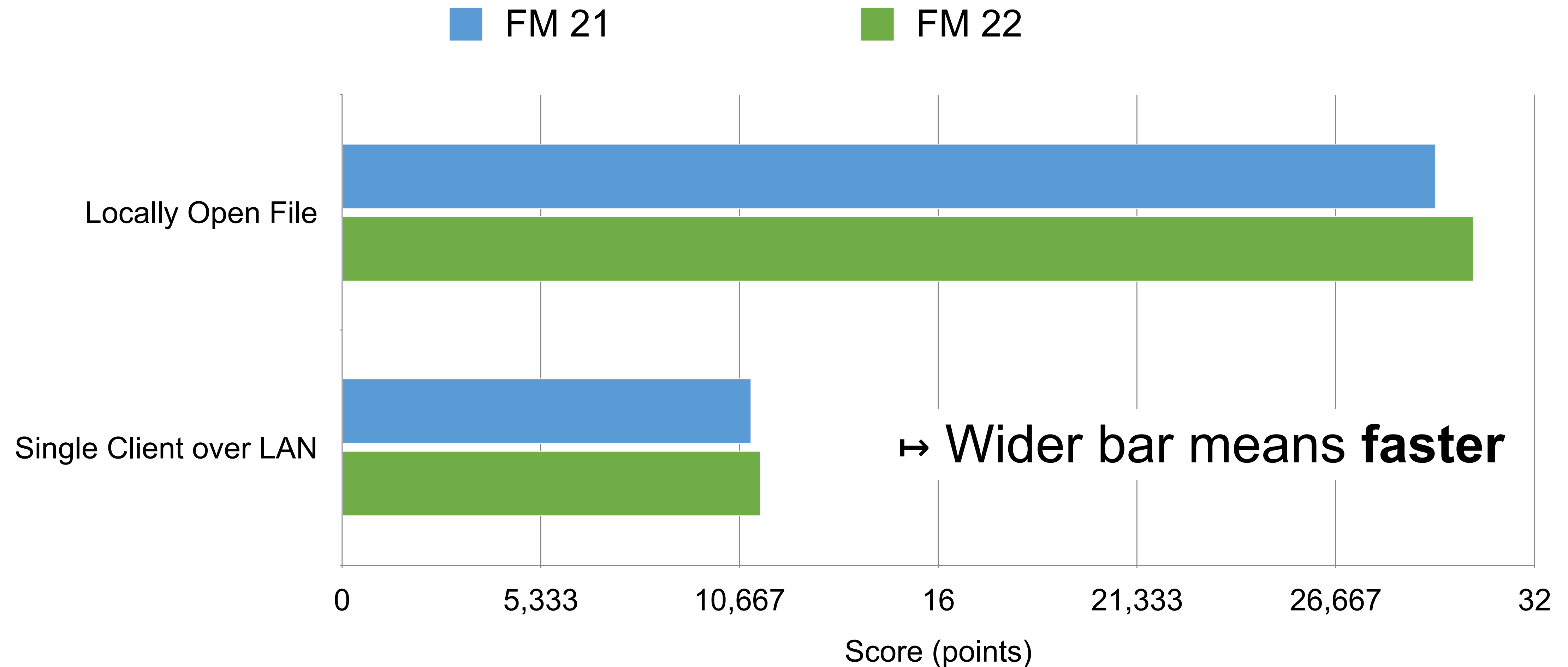
FileMaker 2025 vs 2024

Performance News

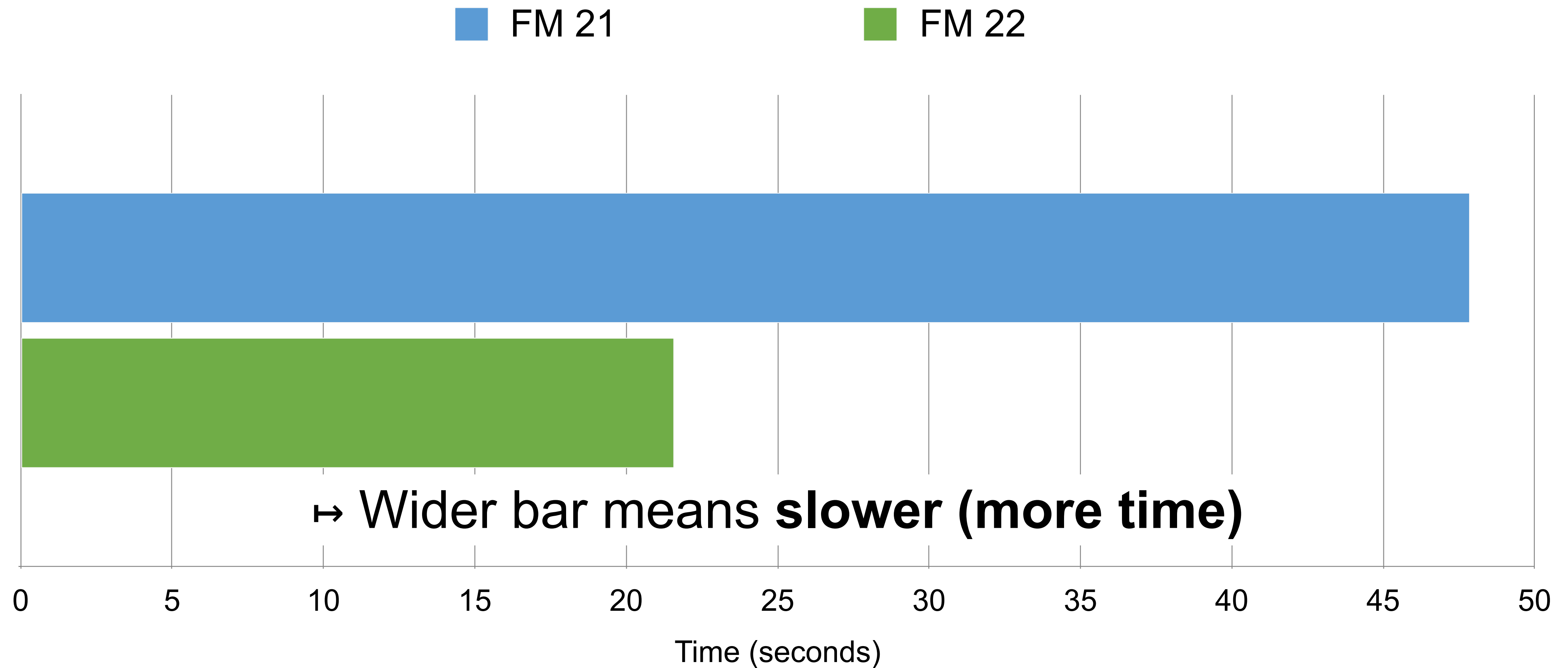
Test Environment

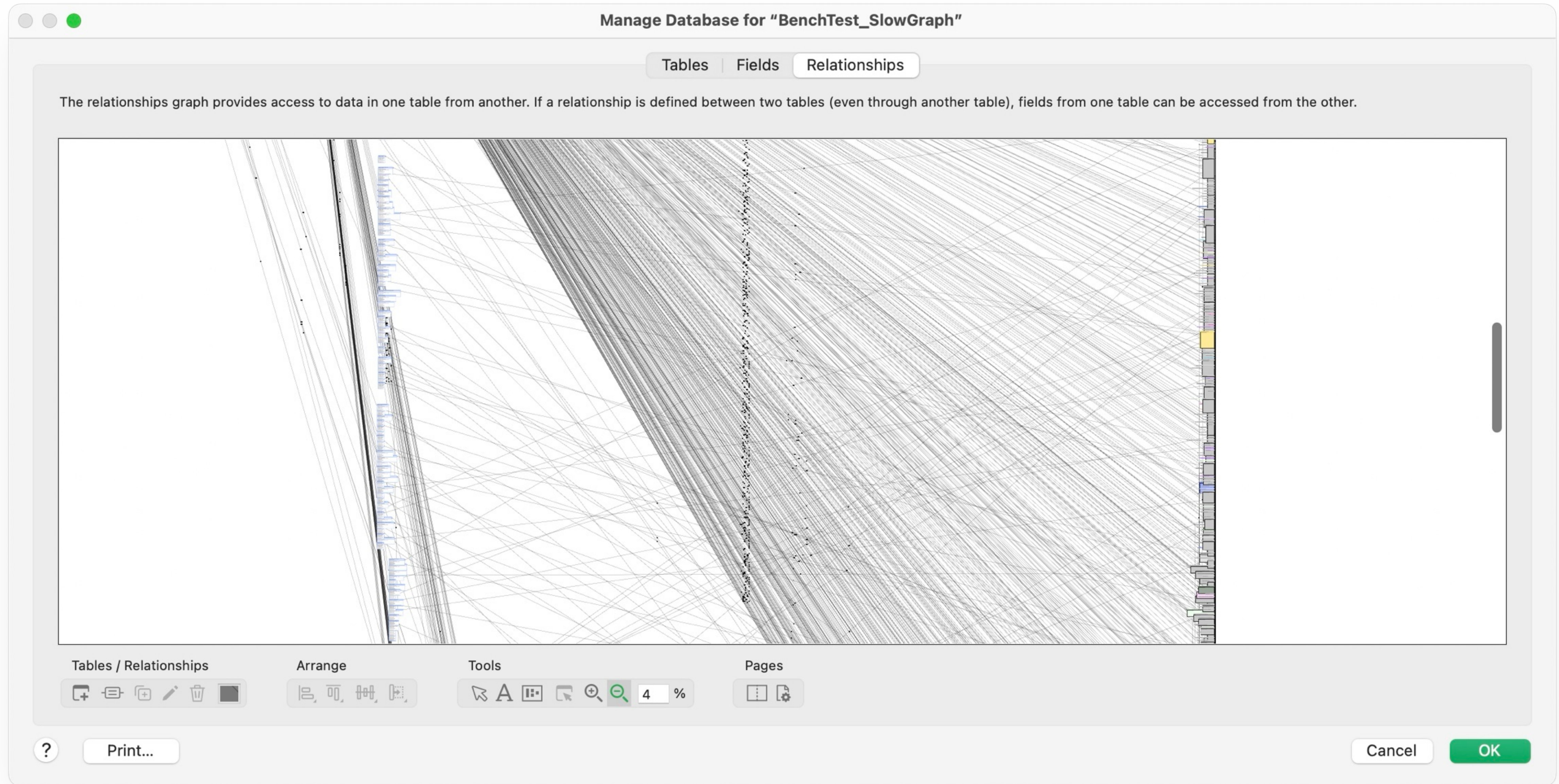
	Server	Client
OS	Ubuntu 22.04.5 LTS	macOS 15.5 Sequoia
Computer model	CompuLab Intense-PC 2012	MacBook Pro
Procesor	Intel Celeron 1.1GHz	Apple M4 Max
RAM	2 GB	128 GB
System HD	100 GB	1 TB
FileMaker 2025	FMS 21.1.6.603	FMP 21.1.1.41
FileMaker 2024	FMS 22.0.2.204	FMP 22.0.1.68

BenchTest - Standard Test (overall performance)

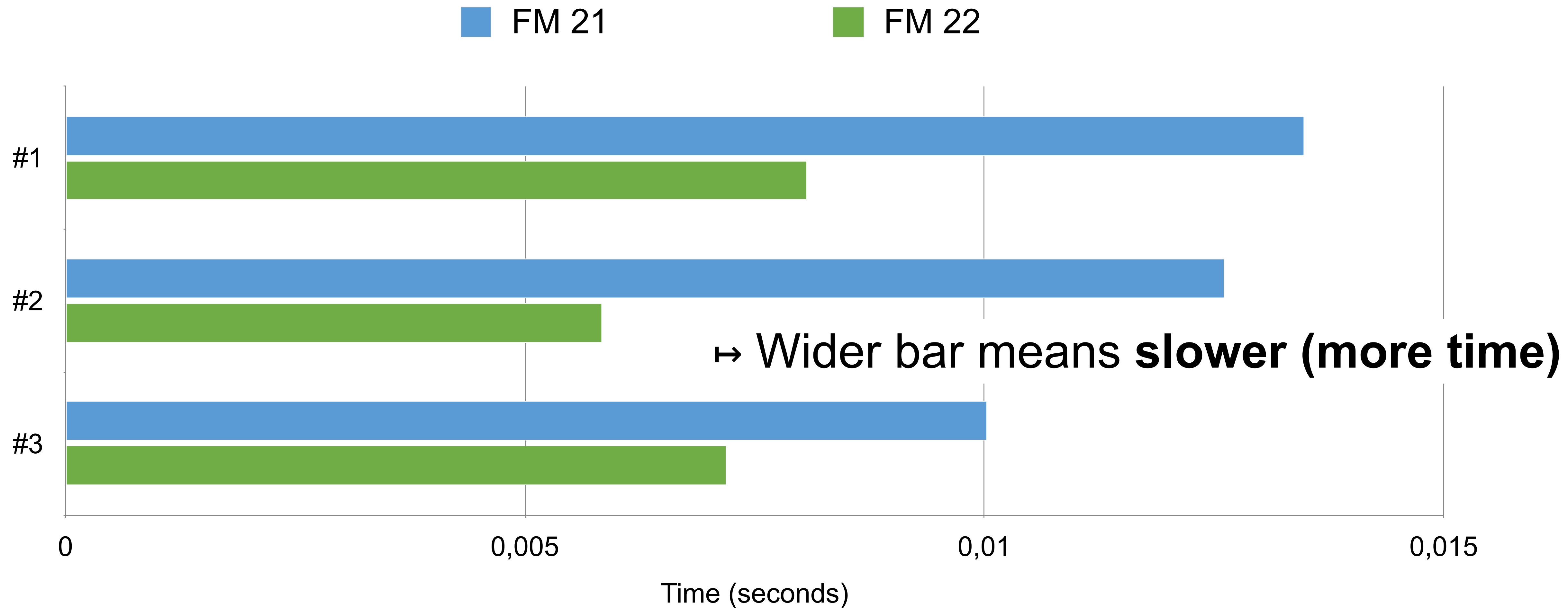


BenchTest - Sort 10,000 records by unindexed unique text

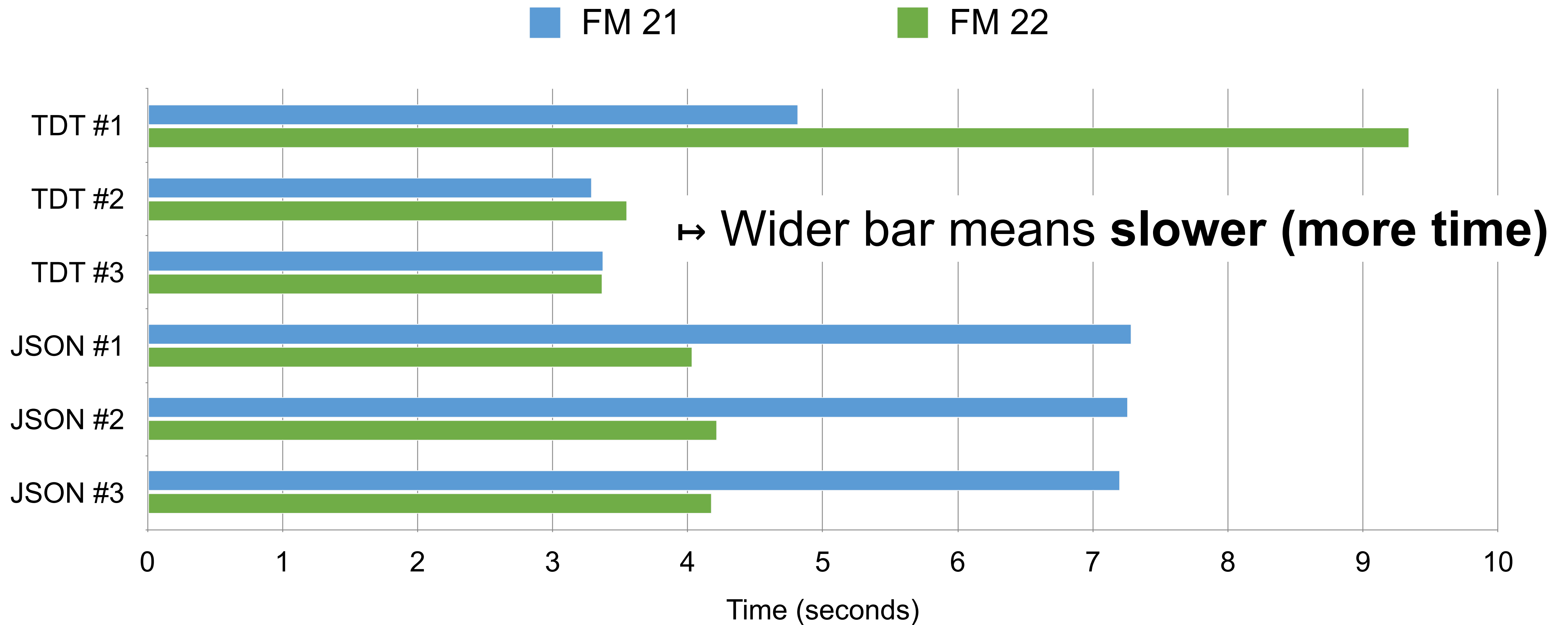




BenchTest - Slow Relationship Graph - File Closing



BenchTest - Sorted Virtual List of 1,000 records



SortJSONArray

```
// This function sorts JSON array of objects based on object value specified by sortKey,
// comparing the keys using the specified locale.
// The main idea of this function is that only operations whose asymptotic complexity is O(n)
// are handled by While loop, while the sorting itself is handled by the SortValues function.

Let ( [

// Count array keys
num_of_records = ValueCount ( JSONListKeys ( json ; "" ) ) ;

// Extract sort keys and their indexes (assuming the values do not contain characters with code < 10)
sortkeys = While ( [ i = 0 ; r = "" ] ; i < num_of_records ; [
r = List ( r ; JSONGetElement ( json ; "[" & i & "]" & sortKey ) & Char(9) & i ) ; i = i + 1
] ; r ) ;

// Sort the keys natively
sortkeys = SortValues ( sortkeys ; 1 ; locale )

// Build re-sorted json
] ; While ( [ i = 1 ; r = "" ] ; i ≤ num_of_records ; [ r = r & "," & JSONGetElement ( json ; 0 + RightWords
( GetValue ( sortkeys ; i ) ; 1 ) ) ; i = i + 1 ] ; Replace ( r ; 1 ; 1 ; "[" ) & "]" )

)
```

BenchTest 2.1 - Initiative '24 & Calculation Engine

BenchTest 2.1

What's faster?
How would you **know**, until you **know**?

About 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 Auto Results **Initiative '24**

Meet Initiative '24

In January 24, 2024, we introduced Initiative '24, our endeavor to make FileMaker calculation engine faster for everyone. Our goal was to convince Claris to make the calculation engine at least 24 times faster by the end of 2024. We're not there yet but we are not giving up...

[Learn more...](#)

Where are we now? What's the next step?

Making FileMaker calculations faster for everyone cannot be done without Claris making the calculation engine faster. So the first step is to convince Claris, as a community, to make it a priority. Here's how we're doing so far:

3840 out of 6620
278 votes (2780 points) to reach target

This is how many votes our idea has got at Claris Community, compared to the #1 top voted idea on the list. Have you voted as well?

[Vote now...](#) [Click here to test your own copy of FileMaker Pro](#) [» How fast is this version?](#)

24U SOFTWARE [Visit Website...](#) [See More Tools...](#) [Close](#)

Copyright © 2013-2025 24U Software. All rights reserved. www.24uSoftware.com Version 2.1v3

BenchTest 2.1 - Initiative '24 & Calculation Engine

BenchTest 2.1

How

About 0 1 2 3 4 5 6 7 8 9 10 11 12 13

Meet Initiative '24

In January 24, 2024, we introduced Initiative '24, our endeavor to make FileMaker calculation engine at least 24 times faster by the end of 2024. We're not there yet but we are not giving up.

[Learn more...](#)

Where are we now? What's the next step?

Making FileMaker calculations faster for everyone cannot be done without Claris making the calculation engine a priority. Here's how we're doing so far:

3840 out of 6620

278 votes (2780 points) to reach target

This is how many votes our idea has got at Claris Community, compared to the #1 top voted idea.

[Vote now...](#) [Click here to test your own copy](#)

24U SOFTWARE

Copyright © 2013-2025 24U Software. All rights reserved.

BenchTest 2.1

What's faster?
How would you know, until you know?

About 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 Auto Results Initiative '24

Calculation Engine Performance

Test Name	Speed (Cycles per Second)		
	Stored	Latest	
Initiative '24 default math speed test	177932	174229	1.02 x slower
Initiative '24 default text speed test	149400	145334	1.03 x slower
Building JSON Array in local script variable	2667	6063	2.27 x faster
Accessing JSON Array Elements in local script variable	22234	257314	11.57 x faster
Building JSON Array in global variable	2654	607	2.29 x faster
Accessing JSON Array Elements in global variable	22193	256179	11.54 x faster
Issue with Position reported by Alex Zueiv in 2022 (see https://community.claris.com/en/s/feed/0D53w00005qdfMaCAI)	7	7	about same
Performance of the previous calculation should be about the same as of this one	338988	337230	1.01 x slower
Issue with FilterValues published by Kevin Frank in May 2020 and attributed to field reference by Alex Zueiv in Dec 2021 (see https://community.claris.com/en/s/feed/0D53w00005qdfMaCAI)	97	96	1.01 x slower
Performance of the previous calculation should be the same or better than of this one	358	357	about same
Accessing JSON Array Elements in pre-parsed local script variable	22343	331371	14.83 x faster

Use the test calculations above to measure performance of one FileMaker Pro version. Then store the collected values, re-open BenchTest on another version, and run the test again.

[Store...](#) [Run all...](#)

24U SOFTWARE

Copyright © 2013-2025 24U Software. All rights reserved.

[Visit Website...](#) [See More Tools...](#) [Close](#)

www.24uSoftware.com Version 2.1v3

Calculation to test in a While loop for 1 second to count cycles

SetRecursion (While ([] = 0 ;

Initialization

```
$result = "" ; $array = ["x77v9dkVWDvcYZWg\","inUneuMlwpCttV9W\","WfxA7AYRYhw2bGtZ\","tpInvrWe6WtSnGKN\","Xsmy9YISh0txpGR6\","E4Q7vxywfoMnJPTv\","bpBAEmHClKJJrPxW\","DEVHpwEF5xm4LZM7\","3qOo0hVA7jcqxdwV\","M85ceS6pAMjQRip\","0t6U40YDXlYwOhli\","4pVHQubOdFqpfzUu\","YJze7F5GG8fgUgio\","iUCoyYU2PRGmUp5B\","EBD84HzNYx0x8c8u\","hClaxzAmeWpseJfG\","hggG1EUybivxhArL\","WaDWB75IRok6Gato\","g33FytVj8CFJzb2b\","IXPpHsRp2ncw38tb\","
```

```
t = Get ( CurrentTimeUTCMicroseconds ) + 1000000  
]; Get ( CurrentTimeUTCMicroseconds) < t ; [
```

Calculation

```
random_index = Int ( 1000 * Random ) ;  
$result = JSONGetElement ( $array ; random_index ) ;
```

```
] = j+1 ] ; j ) ; 100000000 )
```

Database fields to reference in the calculation (for testing field reference impact)

Calculations::data1

Calculations::data2

What's faster?

know, until you know?

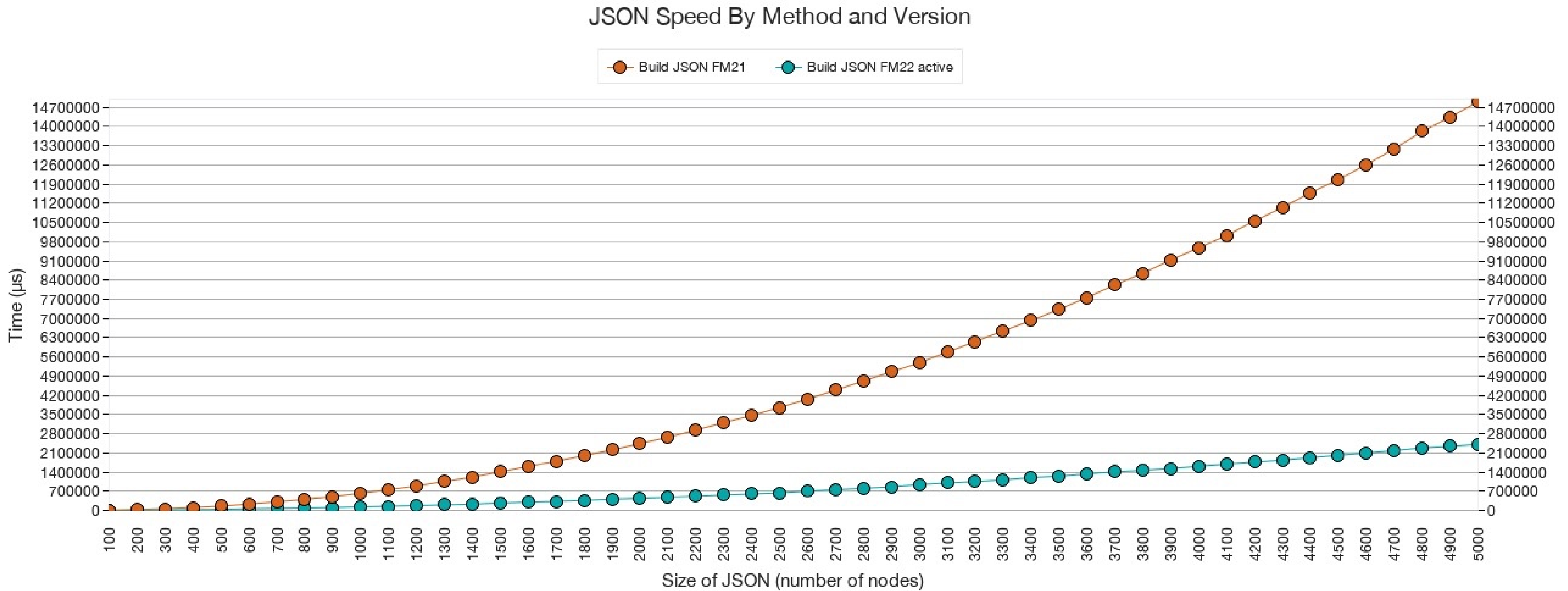
Initiative '24			
Speed (Cycles per Second)			
	Stored	Latest	
▶	177932	174229	1.02 x slower
▶	149400	145334	1.03 x slower
▶	2667	6063	2.27 x faster
▶	22234	257318	11.57 x faster
▶	2654	6074	2.29 x faster
▶	22193	256179	11.54 x faster
▶	7	7	about same
▶	338988	337230	1.01 x slower
▶	97	96	1.01 x slower
▶	358	357	about same
▶	22343	331371	14.83 x faster



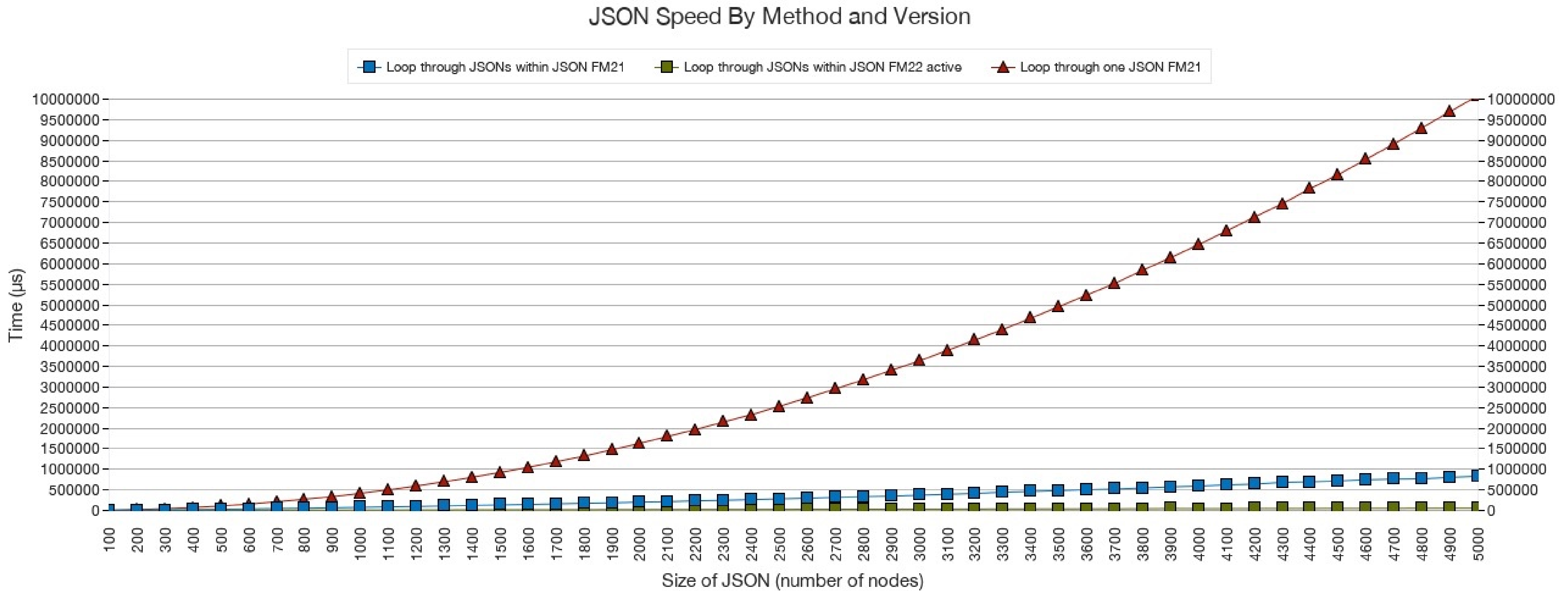
Store...

Run all...

JSON Functions Performance as Function of JSON Size



JSON Functions Performance as Function of JSON Size



JSON Functions Performance as Function of JSON Size

Function	Number of JSON Nodes	Time to complete (seconds)		Times faster
		FileMaker 21	FileMaker 22	
Build JSON	100	0,008281	0,003185	2,6
	1000	0,626694	0,131020	4,8
	5000	14,904656	2,416126	6,2
Loop through JSONs within JSON	100	0,002988	0,001094	2,7
	1000	0,074400	0,010723	6,9
	5000	0,832688	0,053918	15,4
Loop through one JSON	100	0,005440	0,002060	2,6
	1000	0,416500	0,038563	10,8
	5000	10,080045	0,359694	28,0

JSON Functions Performance as Function of JSON Size

Function	Number of JSON Nodes	Time to complete (seconds)		Times faster
		FileMaker 24	FileMaker 22	
Build JSON	100	0,008281	0,003185	2,6
	1000	0,62669	0,131020	4,8
	5000	4,904656	2,416116	6,2
Loop through JSONs within JSON	100	0,002988	0,001094	2,7
	1000	0,074400	0,010723	6,9
	5000	0,832688	0,053918	15,4
Loop through one JSON	100	0,005440	0,002060	2,6
	1000	0,416536	0,38063	10,8
	5000	10,080045	0,359364	28,0

28 x faster
looping through
one JSON with
5000 nodes

Under the Hood

Active Performance improvements

Demo - Bad

```
1 Set Variable [ $name1 ; Value: JSONGetElement ( Demo::JSON1 ; "name" ) ]
2 Set Variable [ $name2 ; Value: JSONGetElement ( Demo::JSON2 ; "name" ) ]
3 Set Variable [ $state1 ; Value: JSONGetElement ( Demo::JSON1 ; "state" ) ]
4 Set Variable [ $state2 ; Value: JSONGetElement ( Demo::JSON2 ; "state" ) ]
```

No help from DBData or Cache

Demo - Better

```
1 Set Variable [ $name1 ; Value: JSONGetElement ( Demo::JSON1 ; "name" ) ]
2 Set Variable [ $state1 ; Value: JSONGetElement ( Demo::JSON1 ; "state" ) ]
3 Set Variable [ $name2 ; Value: JSONGetElement ( Demo::JSON2 ; "name" ) ]
4 Set Variable [ $state2 ; Value: JSONGetElement ( Demo::JSON2 ; "state" ) ]
```

Help from Cache

Demo - Best

```
1 Set Variable [ $json1 ; Value: JSONParse ( Demo::JSON1 ) ]
2 Set Variable [ $json2 ; Value: JSONParse ( Demo::JSON2 ) ]
3
4 Set Variable [ $name1 ; Value: JSONGetElement ( $json1 ; "name" ) ]
5 Set Variable [ $name2 ; Value: JSONGetElement ( $json2 ; "name" ) ]
6 Set Variable [ $state1 ; Value: JSONGetElement ( $json1 ; "state" ) ]
7 Set Variable [ $state2 ; Value: JSONGetElement ( $json2 ; "state" ) ]
```

Pure Native JSON

JSONParseCompatible (json)

```
Let (
  $$JSONParseInvalid =
  Case ( IsEmpty ( $$JSONParseInvalid ) ;
    EvaluationError ( Evaluate ( "JSONParse ( \"{}\") " ) ) ;
    $$JSONParseInvalid ) ;
  Case ( $$JSONParseInvalid ; json ; JSONParse ( json ) )
)
```

You Can Avoid JSONParse

All these functions
return parsed JSON

The screenshot shows the 'Edit Expression' dialog in FileMaker. The 'Expression:' field contains a list of functions: `List (JSONParsedState ("{}") ; JSONParsedState (JSONParse ("{}")) ; JSONParsedState ("" & JSONParse ("{}")) ; JSONParsedState (JSONFormatElements ("{}")) ; JSONParsedState (JSONSetElement ("{}" ; "a" ; "" ; JSONString)) ; JSONParsedState (JSONGetElement ("{}" ; "")) ; JSONParsedState (JSONDeleteElement ("{\\"a\\":1}" ; "junk")) ; JSONParsedState (JSONDeleteElement ("{\\"a\\":1}" ; "")))`. A red box highlights the last four functions: `JSONFormatElements`, `JSONSetElement`, `JSONGetElement`, and `JSONDeleteElement`. A red arrow points from the text 'All these functions return parsed JSON' to the `JSONParse` function. Another red arrow points from the text 'This one doesn't' to the `JSONDeleteElement` function. The 'Result:' field shows a list of values: 0, 3, 0, 3, 3, 3, 3, 0. The last three values (3, 3, 3) are highlighted in green. The 'Evaluate' button is visible.

Specify a field, variable, or expression to monitor. This expression will be evaluated based on the context determined at runtime.

<unknown> Search

Expression:

```
List (
JSONParsedState ( "{}" ) ;
JSONParsedState ( JSONParse ( "{}" ) ) ;
JSONParsedState ( "" & JSONParse ( "{}" ) ) ;
JSONParsedState ( JSONFormatElements ( "{}" ) ) ;
JSONParsedState ( JSONSetElement ( "{}" ; "a" ; "" ; JSONString ) ) ;
JSONParsedState ( JSONGetElement ( "{}" ; "" ) ) ;
JSONParsedState ( JSONDeleteElement ( "{\\"a\\":1}" ; "junk" ) ) ;
JSONParsedState ( JSONDeleteElement ( "{\\"a\\":1}" ; "" ) )
)
```

Result:

0
3
0
3
3
3
3
0

☐ Automatically evaluate Evaluate

This one doesn't

GetRecordIDsFromFoundSet

titles

Search

id

_created

Tt _created_by

_modified

Tt _modified_by

fx _one

Tt -----

Tt name_mine

Tt name_czech

Tt name_original

Tt medium

Tt format

Expression:

GetRecordIDsFromFoundSet (ValueNumber)

Result:

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

Automatically evaluate

Evaluate

" "

()

&

¶

=

≠

>

<

≥

≤

+

-

/

*

not

and

or

xor

^

GetRecordIDsFromFoundSet

titles

Search

id

_created

Tt _created_by

_modified

Tt _modified_by

fx_one

Tt -----

Tt name_mine

Tt name_czech

Tt name_original

Tt medium

Tt format

Expression:

GetRecordIDsFromFoundSet (ValueNumberRanges)

Result:

1-47
51-53
57-79
81-122
124-146
148
150
152-225
227-235
247
249-253
255-264
268-274
277-280
283-288
292-293
296-304
306-324

Automatically evaluate

Evaluate

" "

()

&

¶

=

≠

>

<

≥

≤

+

-

/

*

not

and

or

xor

^

GetRecordIDsFromFoundSet

titles

Search

id

_created

Tt _created_by

_modified

Tt _modified_by

fx _one

Tt -----

Tt name_mine

Tt name_czech

Tt name_original

Tt medium

Tt format

Expression:

GetRecordIDsFromFoundSet (JSONNumber)

Result:

☐ Automatically evaluate

[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,45,46,47,51,52,53,57,58,59,60,61,62,63,64,65,66,67,68,69,70,71,72,73,74,75,76,77,78,79,81,82,83,84,85,86,87,88,89,90,91,92,93,94,95,96,97,98,99,100,101,102,103,104,105,106,107,108,109,110,111,112,113,114,115,116,117,118,119,120,121,122,124,125,126,127,128,129,130,131,132,133,134,135,136,137,138,139,140,141,142,143,144,145,146,148,150,152,153,154,155,156,157,158,159,160,161,162,163,164,165,166,167,168,169,170,171,172,173,174,175,176,177,178,179,180,181,182,183,184,185,186,187,188,189,190,191,192,193,194,195,196,197,198,199,200,201,202,203,204,205,206,207,208,209,210,211,212,213,214,215,216,217,218,219,220,221,222,223,224,225,227,228,229,230,231,232,233,234,235,247,249,250,251,252,253,255,256,257,258,259,260,261,262,263,264,268,269,270,271,272,273,274,277,278,279,280,283,284,285,286,287,288,292,293,296,297,298,299,300,301,302,303,304,306,307,308,309,310,311,312,313,314,315,316,317,318,319,320,321,322,323,324,326,327,328,329,331,333,334,335,337,338,339,340,341,342,343,344,345,346,347,348,349,350,351,352,353,354,355,356,357,358,359,360,361,362,363,364,365,366,367,368,371,372,373,377,381,388,389,390,391,392,393,394,395,396,397,398,399,402,403,407,408,409,410,411,412,413,414,418,419,422,423,424,425,426,427,428,430,431,432,435,436,439,440,441,442,443,446,447,452,453,454,455,456,457,459,460,461,462,463,464,465,466,467,468,469,470,471,473,474,475,477,478,479,480,481,484,485,486,487,488,489,491,492,493,494,495,496,497,498,499,500,501,502,503,504,506,507,508,509,511,512,513,514,515,516,517,518,519,521,522,523,525,526,529,530,534,538,539,540,542,545,547,548,549,550,553,554,555,556,557,558,559,561,562,565,566,568,569,572,573,
]

GetRecordIDsFromFoundSet

titles

Search

id

_created

Tt _created_by

_modified

Tt _modified_by

fx_one

Tt -----

Tt name_mine

Tt name_czech

Tt name_original

Tt medium

Tt format

Expression:

GetRecordIDsFromFoundSet (JSONString)

Result:

Automatically evaluate

Evaluate

["1","2","3","4","5","6","7","8","9","10","11","12","13","14","15","16","17","18","19","20","21","22","23","24","25","26","27","28","29","30","31","32","33","34","35","36","37","38","39","40","41","42","43","44","45","46","47","51","52","53","57","58","59","60","61","62","63","64","65","66","67","68","69","70","71","72","73","74","75","76","77","78","79","81","82","83","84","85","86","87","88","89","90","91","92","93","94","95","96","97","98","99","100","101","102","103","104","105","106","107","108","109","110","111","112","113","114","115","116","117","118","119","120","121","122","124","125","126","127","128","129","130","131","132","133","134","135","136","137","138","139","140","141","142","143","144","145","146","148","150","152","153","154","155","156","157","158","159","160","161","162","163","164","165","166","167","168","169","170","171","172","173","174","175","176","177","178","179","180","181","182","183","184","185","186","187","188","189","190","191","192","193","194","195","196","197","198","199","200","201","202","203","204","205","206","207","208","209","210","211","212","213","214","215","216","217","218","219","220","221","222","223","224","225","227","228","229","230","231","232","233","234","235","247","249","250","251","252","253","255","256","257","258","259","260","261","262","263","264","268","269","270","271","272","273","274","277","278","279","280","283","284","285","286","287","288","292","293","296","297","298","299","300","301","302","303","304","306","307","308","309","310","311","312","313","314","315","316","317","318","319","320","321","322","323","324","326","327","328","329","331","333","334","335","337","338","339","340","341","342","343","344","345","346","347","348","349","350","351","352","353"]

GetRecordIDsFromFoundSet

titles

Search

id

_created

Tt _created_by

_modified

Tt _modified_by

fx _one

Tt -----

Tt name_mine

Tt name_czech

Tt name_original

Tt medium

Tt format

Expression:

GetRecordIDsFromFoundSet (JSONStringRanges)

Result:

☐ Automatically evaluate

["1-47","51-53","57-79","81-122","124-146","148","150","152-225","227-235","247","249-253","255-264","268-274","277-280","283-288","292-293","296-304","306-324","326-329","331","333-335","337-368","371-373","377","381","388-399","402-403","407-414","418-419","422-428","430-432","435-436","439-443","446-447","452-457","459-471","473-475","477-481","484-489","491-504","506-509","511-519","521-523","525-526","529-530","534","538-540","542","545","547-550","553-559","561-562","565-566","568-569","572-573","575-576","578","580","582","586","588","593-596","599-601","604-605","611-616","621","629","631-638","640-648","650-660","662-670","672","675","677","682-710","713-716","718","721-729","732","734-746","750-751","754-758","762","766-771","773-779","782-797","799-824","826-839","841-851","857-928","930-981","984-993","995-1006","1011-1027","1029-1033","1035-1106","1108-1146","1148-1198","1204","1206","1208-1210","1326-1327","1335-1357","1359-1362","1364-1397"]

Go to List of Records

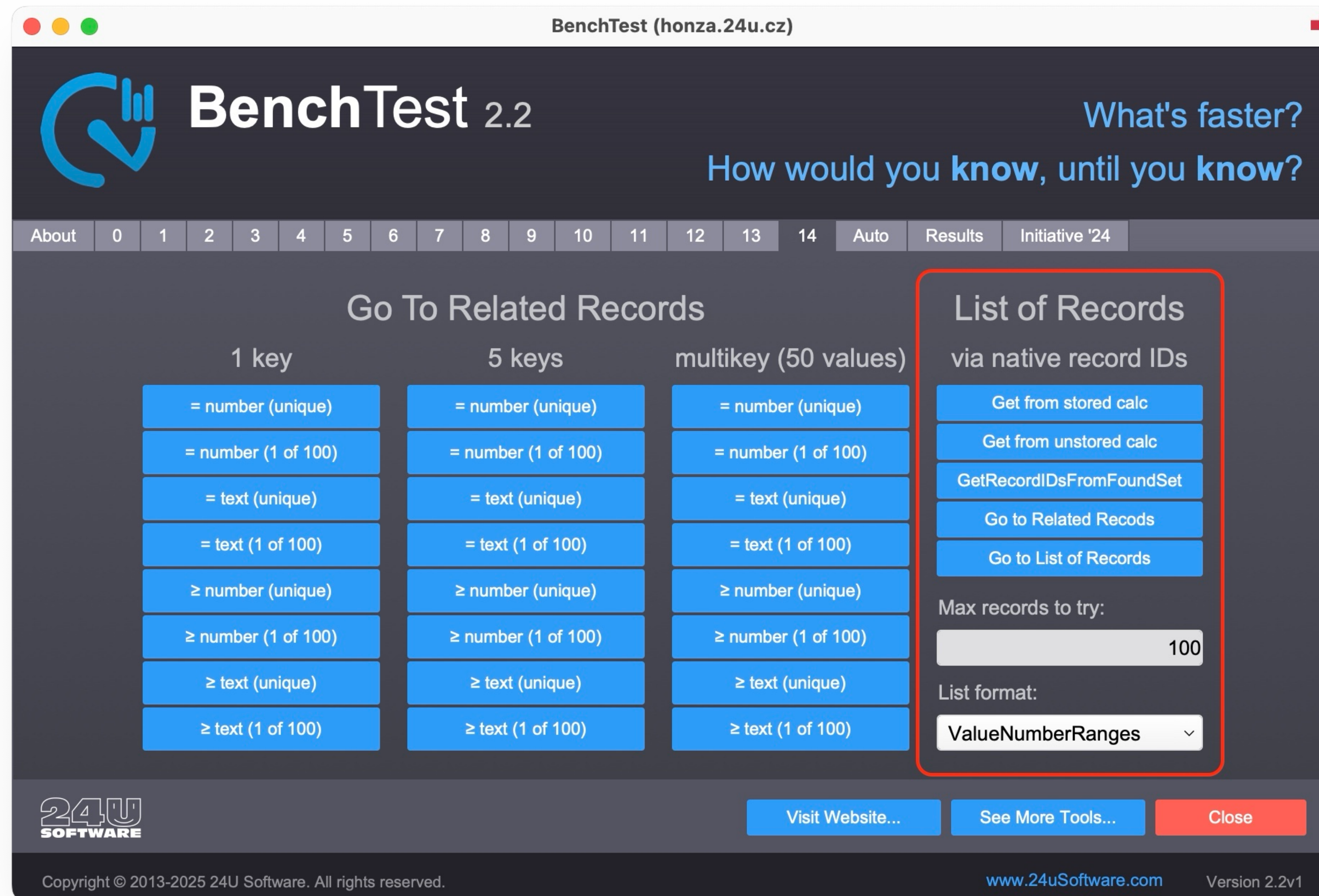
Edit Script "Test 14 D4 core (GTLR from <<\$max_records>> records by GTRR)" (BenchTest (honza.24u.cz))

```
1  
2 Set Field [ Test14::g_key1 ; Get ( ScriptParameter ) ]  
3 Go to Related Record [ Show only related records ; From table: "Test14_Multikey_RecordID" ;  
4 Using layout: "X blank form" (X) ]  
5 Exit Script [ Text Result: ]
```

Edit Script "Test 14 D5 core (GTLR from <<\$max_records>> records by GTLR as <<\$variant>>)" (BenchTest (h...))

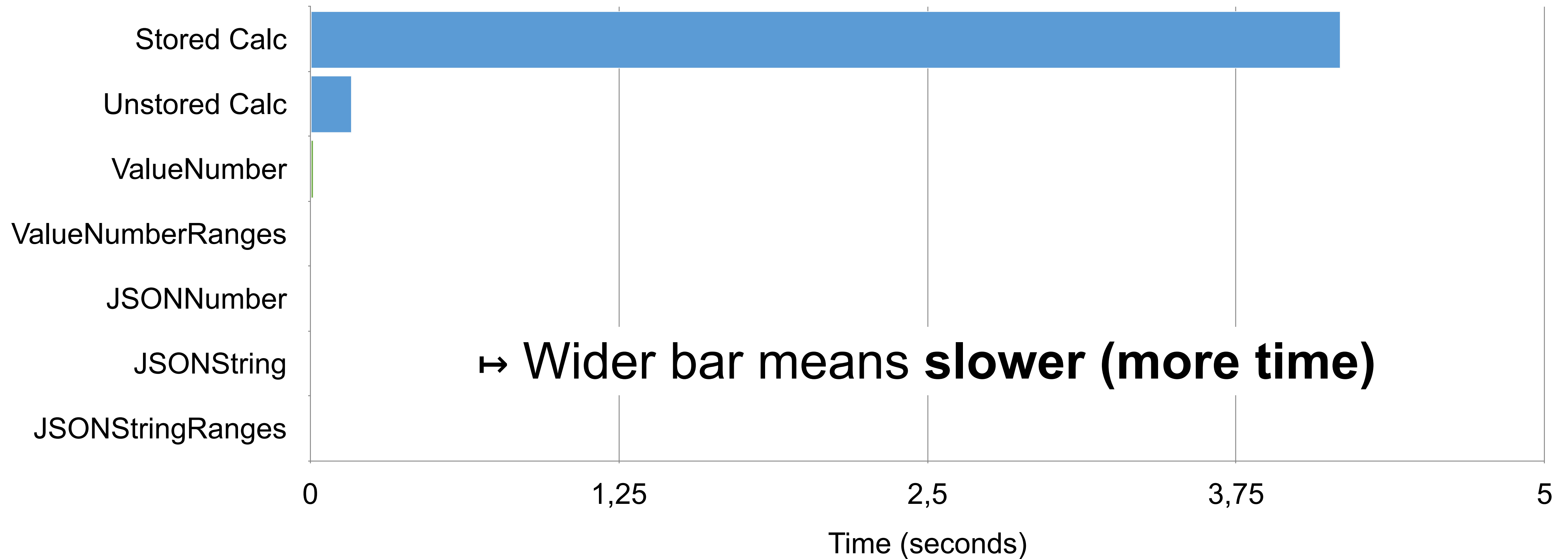
```
1  
2 Go to List of Records [ List of record IDs: Get ( ScriptParameter ) ; Using layout: "X blank form" (X) ;  
3 Animation: None ]  
4 Exit Script [ Text Result: ]
```

BenchTest 2.2 - Go to List of Records



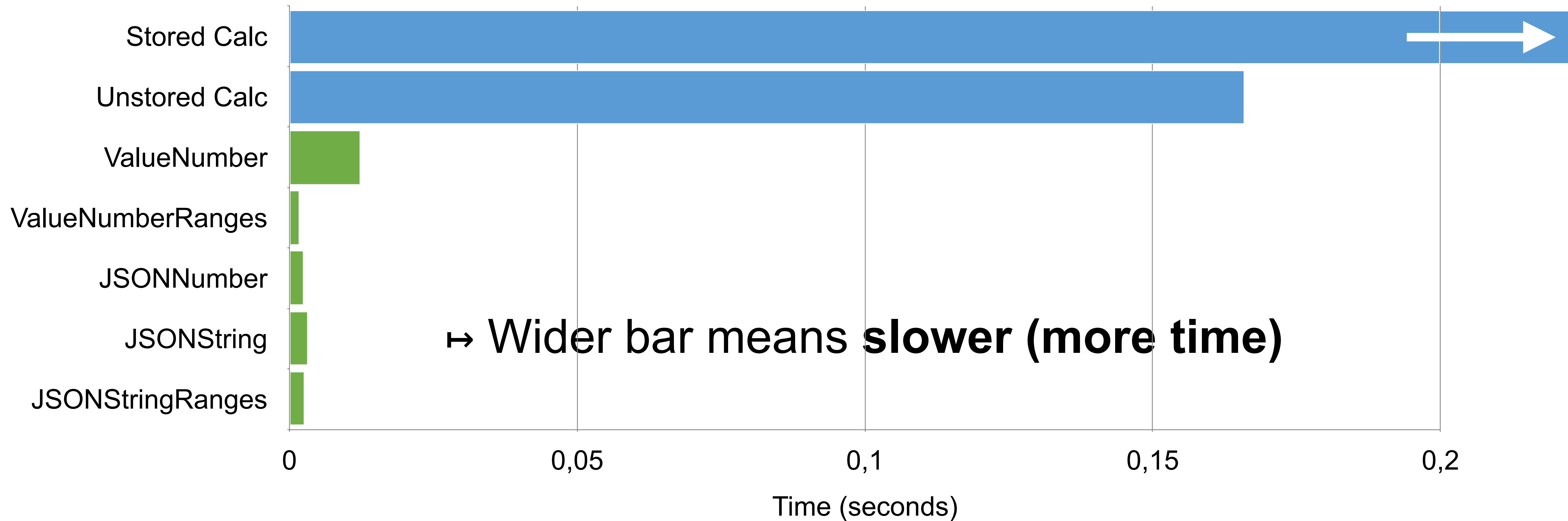
Storing Found Set

■ List(cRecordID) ■ GetRecordIDsFromFoundSet



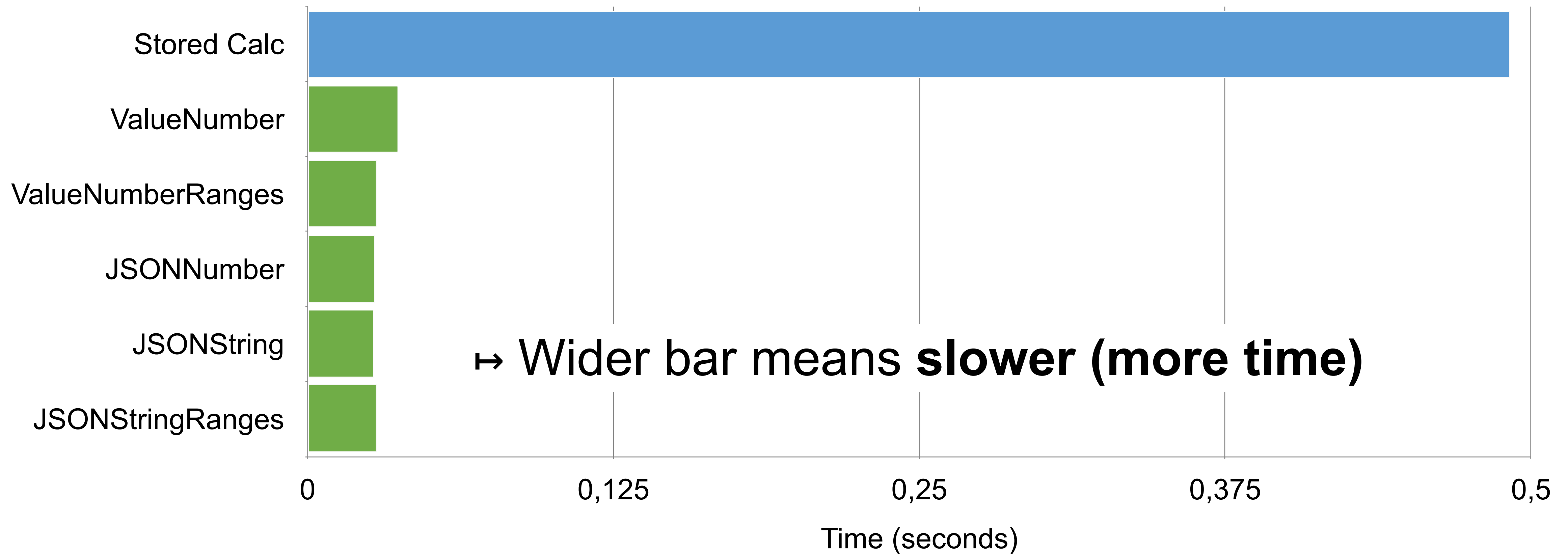
Storing Found Set

List(cRecordID) GetRecordIDsFromFoundSet



Restoring Found Set

■ Go to Related Records ■ Go to List of Records



Now let's make all other functions faster as well...



Vote here



278 votes (2780 points) to reach target



Learn why

Even if you have voted already,
you certainly know someone who has not...
...so you can still help

Q & R

Vielen Dank für Ihr Interesse!

Vielen Dank unseren Sponsoren und Konferenz-Partnern

