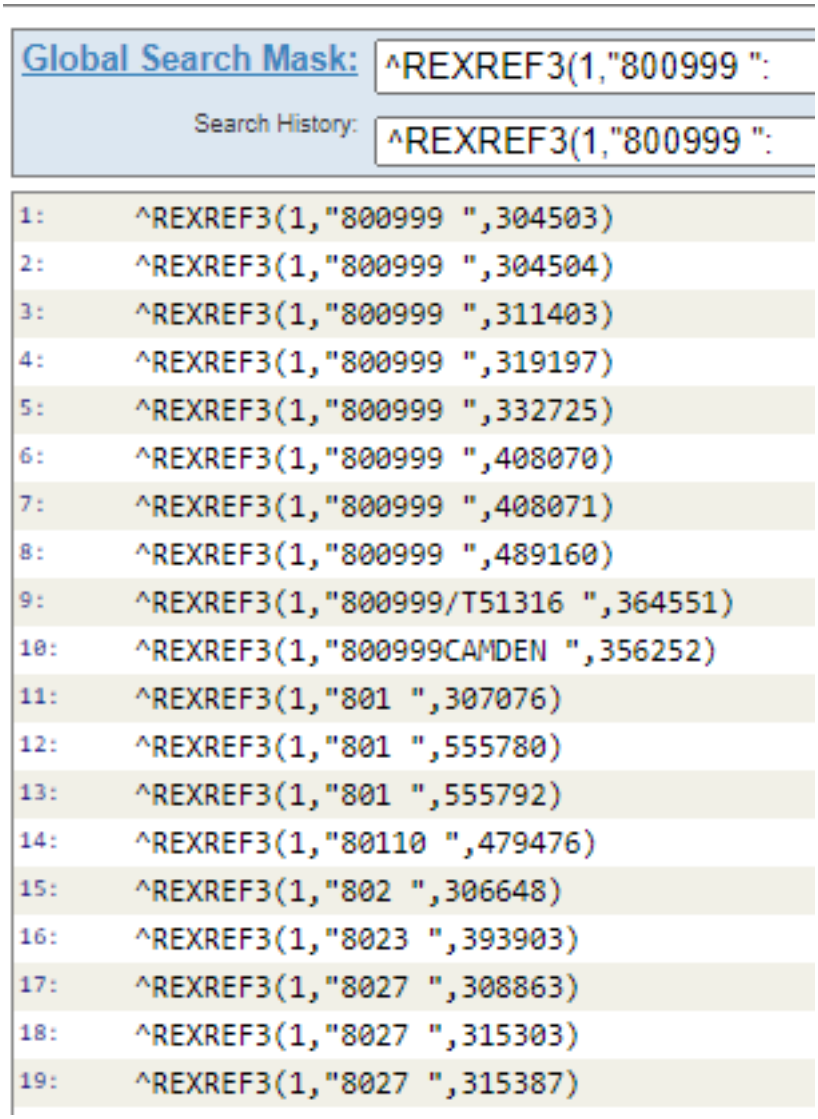
Question
Mark Charlton · Feb 17, 2023

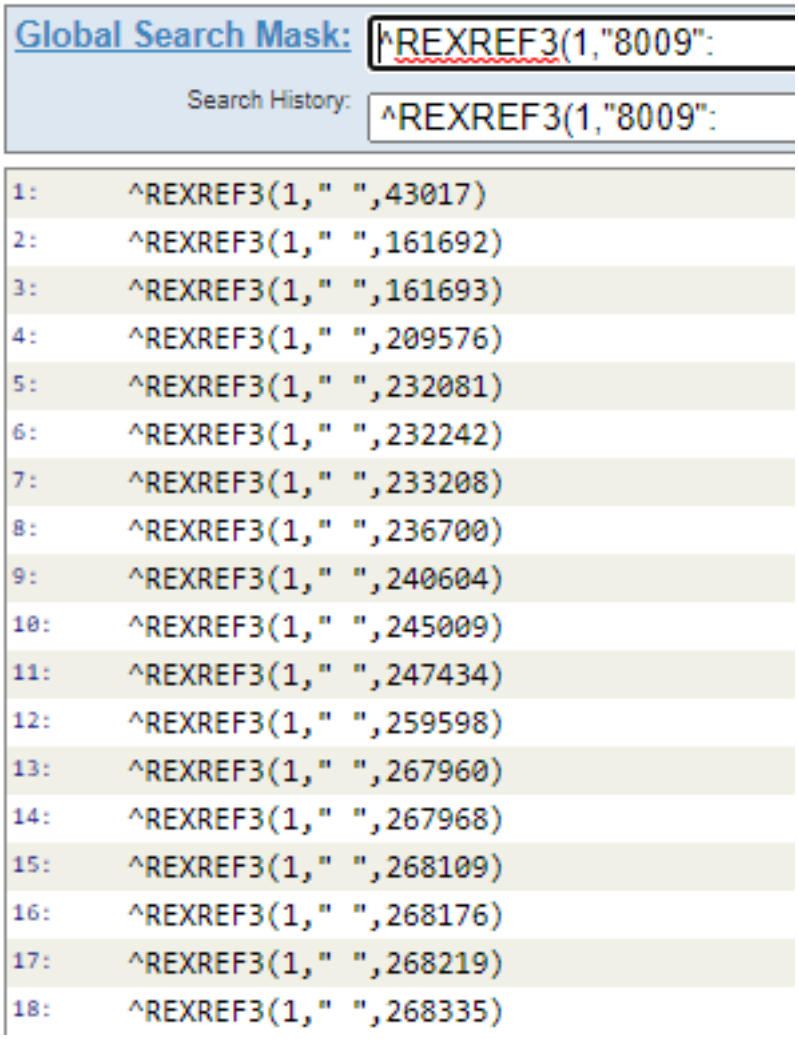
Global View and $ORDER not returning expected search results, (or what I want it to).

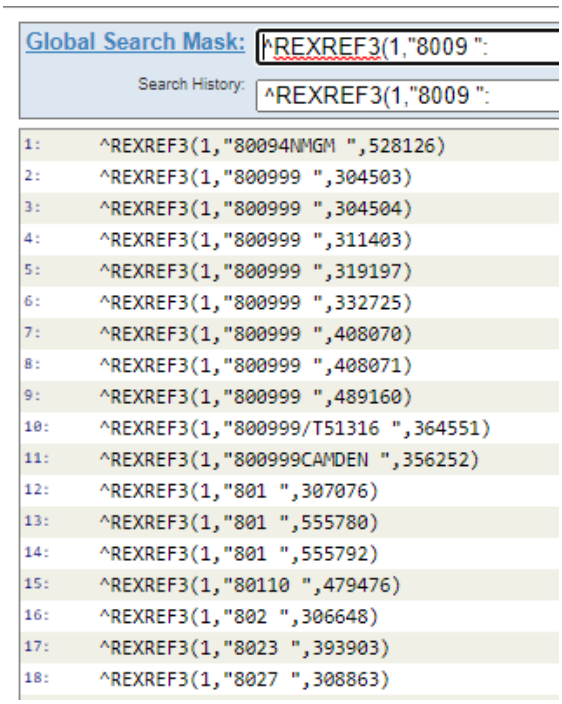
I know this problem something to do with sort and collation and string comparison vs numeric only strings, but I can't fathom out the details of it.

I have the following global, where all values are suffixed with a space, I believe the intention is to prevent empty strings and force string comparisons rather than numeric. Which is fine when searching for an exact match, so searching for "ABC" internally looks for "ABC ", or "800999" looks for "800999 ". However one function tries to search for strings containing, (specifically beginning with), a partial code. So in the below data the example could be looking for prefixes 8009.

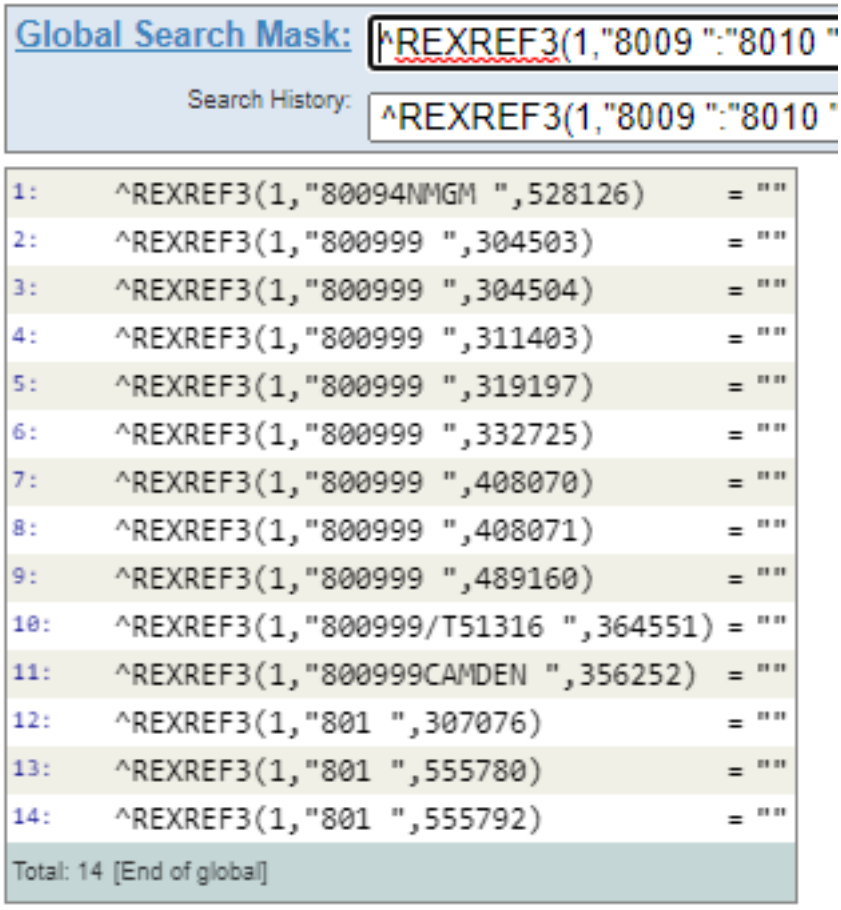


Trying that in management portal using either "8009" or "8009 " gives different results and I don't fully understand how "8009 ": matches "800999 " but "8009": doesn't. I'm assuming that a pure number sorts before any strings, which means "8009" matches the string " " and if I kept paging through the entire data I'd come to the data I wanted. But that's loading and returning far more data than I want to process.


Global Search Mask:
^REXREF3(1,"8009":
Search History:
^REXREF3(1,"8009":
1: ^REXREF3(1," ",43017)
2: ^REXREF3(1," ",161692)
3: ^REXREF3(1," ",161693)
4: ^REXREF3(1," ",209576)
5: ^REXREF3(1," ",232081)
6: ^REXREF3(1," ",232242)
7: ^REXREF3(1," ",233208)
8: ^REXREF3(1," ",236700)
9: ^REXREF3(1," ",240604)
10: ^REXREF3(1," ",245009)
11: ^REXREF3(1," ",247434)
12: ^REXREF3(1," ",259598)
13: ^REXREF3(1," ",267960)
14: ^REXREF3(1," ",267968)
15: ^REXREF3(1," ",268109)
16: ^REXREF3(1," ",268176)
17: ^REXREF3(1," ",268219)
18: ^REXREF3(1," ",268335)



I can limit the data manually by putting in an end range on management portal because I can look at the data and assess how to end range it.



however if I was to do that with a $O in code where the search prefix is user supplied I can't calculate the end range in code to limit the options. So I end up having to search through the entire global from the point of the match forwards. (I mean I suppose I could calculate ascii values of end characters and increment etc or TR replace the next character but I don't want to be getting into that sort of messy code unless I absolutely have to).

This is my test fragment of code where I found the error manifesting.

```
findPOs
// test params
k ^WK3SORT($J)
S ^WK3SORT($J,"800999 ")=1
S ^WK3SORT($J,"800999")=1
S ^WK3SORT($J,"8009")=1
S ^WK3SORT($J,"8009 ")=1
s %C=1
s loopCountTest=0
s loopCountSearch=0
w "Testing PO Search",! SORT3 //Find all purchase orders that starts with the target string
N POREF,LOOPREF
S LOOPREF=""
S3L1 S LOOPREF=$O(^WK3SORT($J,LOOPREF)) G:LOOPREF="" S3L1X
 s loopCountSearch=0
s loopCountTest=loopCountTest+1
w "Searching for :@"_LOOPREF_"@: ",!
S (TargetPORef,POREF)=$$SORTCASE^UTL00EXT(LOOPREF) //"_"
S3L2 S POREF=$O(^REXREF3(%C,POREF)) G:POREF="" S3L2X
s loopCountSearch=loopCountSearch+1
// If its not a match, continue searching
G:($E(POREF,1,$L(TargetPORef))'=TargetPORef) S3L2
w "Match: "_POREF_" CONNO: "
S CONNO=""
S3L3 S CONNO=$O(^REXREF3(%C,POREF,CONNO)) G:CONNO="" S3L3X
w CONNO_","
G S3L3
S3L3X
w !
G S3L2
S3L2X //
W "Searched : " _ loopCountSearch,!
G S3L1
S3L1X //
W "Tested : " _ loopCountTest,!
Q
```

You can see below that the output is correct, however the searched number is huge, (for the number of records I need).

```
Testing PO Search
Searching for :@8009@:
Match: 80094NMGM  CONNO: 528126,
Match: 800999  CONNO: 304503,304504,311403,319197,332725,408070,408071,489160,
Match: 800999/T51316  CONNO: 364551,
Match: 800999CAMDEN  CONNO: 356252,
Searched : 161187
Searching for :@800999@:
Match: 800999  CONNO: 304503,304504,311403,319197,332725,408070,408071,489160,
Match: 800999/T51316  CONNO: 364551,
Match: 800999CAMDEN  CONNO: 356252,
Searched : 161187
Searching for :@8009 @:
Match: 80094NMGM  CONNO: 528126,
Match: 800999  CONNO: 304503,304504,311403,319197,332725,408070,408071,489160,
Match: 800999/T51316  CONNO: 364551,
Match: 800999CAMDEN  CONNO: 356252,
Searched : 161187
Searching for :@800999 @:
Match: 800999  CONNO: 304503,304504,311403,319197,332725,408070,408071,489160,
Match: 800999/T51316  CONNO: 364551,
Match: 800999CAMDEN  CONNO: 356252,
Searched : 161187
Tested   : 4
```

Previously the function above was as follows:

```
S3L2 S POREF=$O(^REXREF3(%C,POREF)) G:POREF=""!($E(POREF,1,$L(TargetPORef))'=
TargetPORef) S3L2X
```

where the string match exited the loop if the first results weren't a string match. The problem there is that this didn't work for fully numeric values, which is why I had to adapt it to the function as seen above. I tried just adding the training space onto the search parameter but that didn't return any values at all (you can see where its commented out in the function).

So, dear hive mind, what obvious piece of information am I missing?

#Globals #Key Question #ObjectScript #Caché
Product version: Caché 2018.1
$ZV: 2018.1.4.505.1

Source URL:https://community.intersystems.com/post/global-view-and-order-not-returning-expected-search-results-or-what-i-want-it