%PARALLEL Query Hint (2016)

Brendan Bannon – Support Manager

March 09 2016

%PARALLEL

- Use multiple CPUs to resolve a single SQL statement
- Works best with queries that process a large quantity of data and return a small result set
  - Aggregate
  - GROUP BY
- %PARALLEL can be used with FROM or UNION
- %PARALLEL is only applied at the highest query level in which it is specified. (2015.2)
- %PARALLEL is ignored when…….list is long but getting shorter.
- 2015.2, 2016.1 and 2016.2 stay away from UNION %PARALLEL

Shared Memory Heap - MAK4367

First in 2016.2

- We now use IPQ for parallel SQL code, this is shared memory from SMH so parallel SQL will put additional pressure on the SMH size and customers may need to increase SMH size if they will be using parallel SQL. The rule of thumb for this is each IPQ is 4x64k=256k and we will split a SQL query into the number of CPU cores we have available so users need to allocate this much extra SMH:

  <Number concurrent parallel SQL requests> x <Number cores > x 256 = <size in k they need to increase SMH>

  If they fail to do this they will see errors in cconsole.log showing they are out of SMH and SQL queries will fail and they may well get other errors as other subsystems
  try to allocate SMH.

Jobbed Processes - MAK4367

First in 2016.2

- The old work queue worker jobs (2015.2) were jobbed off by the current process, this meant that they are running as the current users OS level user
- Old worker jobs would count toward the 25 process limit (2015.2)
- With the new shared pool of worker jobs these are all running as the ccontrol 'cacheusr' OS level user
- New worker jobs do not take a license slot

You'll make breakthroughs

INTERSYSTEMS®

FROM Hint

Simple FROM example

Show Plan

- SELECT Name, DOB FROM %PARALLEL Sample.Person

- Split master map Sample.Person.IDKEY into ranges of T1.ID values.

- For each range, queue a query for parallel execution:
  SELECT %EXACT T1.Name,T1.DOB
FROM %NOPARALLEL Sample.Person T1 %PARTITION BY [range condition on T1.ID]

- Pipe results into temp-file A

Splitting up the work

New calls for %Parallel

- In the generated INT code there will be a call to
  - ##class(%SQL.Agent).%RunWork(global,start,end,function,args,prefix)
  - In that method we will find the number of CPUs available
    - $$calcJobs^%SYS.WorkQueueMgr()
  - Then we will figure out where to split the data
    - $$SplitSize^%GSIZE(GlobalRef,Mode,Parameter,StartBound,EndBound,Prefix)

Generated INT code

- Routine contains 2 queries: the original and the rewrite that will be run by the agents.
  - SELECT Name, DOB FROM %Parallel Sample.Person
    - generated code is getting data out of an object, not a global
    - Have a look at %0Afirst tag
    - S %cur029732d(8)=%cur029732d(6).Get()
  - SELECT %EXACT T1,Name, T1,DOB, T1,ID FROM %NoParallel Sample.Person T1 %PARTITION BY T1.ID >:start and T1,ID <= :end
    - Generates code that loops over globals
    - Have a look at %0Bfirst tag
    - S %sqlout3=$o(^Sample.PersonD(%sqlout3),1,%sqldata17d)

%SYS.PTools.SQLStats

- Cursor cur0 is the main cursor
  - Run once
- Cursor p0AmBm210t0 is the cursor for the parallel query
  - Run 4 times for 4 different id ranges
- The SQLStats for the cur0 cursor include the stats from the worker job that run p0AmBm210t0
- %Parallel has a Global Ref overhead of 681-(4*51) = 477

Cursor Name	Run Count	Avg Rows	Avg Glorefs	Avg Disk Wait	Avg Commands	Avg Time
cur0	1	200	681	0	17431	.05457
p0AmBm2l0t0	4	50	51	0	1575	.00027

What is going on with all the Global Refs?

^PERFMON with GG

- Globals involved with %Parallel query when ^Test2 is run 1000 times

Global Name	TotRefs	% Refs	GloSet	GloKill	LogBlkRd	Directory
%qCacheObjectQualifier	2868.1	30.4	0	0	77.9	c:\intersystems\cache\latest\mgr\cachelib\
Sample.PersonD	2278.9	24.2	0	0	2363.5	c:\intersystems\cache\latest\mgr\samples\
%ISCWorkQueue	1834.6	19.5	587	33.8	1568.1	c:\intersystems\cache\latest\mgr\cache\
%ISCLOG	844.8	9	0	0	1173.7	c:\intersystems\cache\latest\mgr\
ROUTINE	85.5	0.9	0	0	214	c:\intersystems\cache\latest\mgr\
%ISC.WorkQueueMgr	47.1	0.5	0	0	0	c:\intersystems\cache\latest\mgr\cachetemp\

Debugging Worker Jobs - ^%ISCLOG

- The Worker Job code will log info into ^%ISCLOG if set to 2 or 3

```
^%ISCLOG=3
^%ISCLOG("Data")=76
^%ISCLOG("Data",1)=$lb(2,"WorkMgr","CreateWorkers","4472","SAMPLES","2016-03-16 17:18:39.333","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",1,0)=$lb("enablemulti","1","origgroup",-1,"group","1","ipq",1)
^%ISCLOG("Data",2)=$lb(3,"WorkMgr","CreateWorkers: Created queues","4472","SAMPLES","2016-03-16
17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",2,0)=$lb("q1","^%ISCWorkQueue(1,0)","q2","^%ISCWorkQueue(1,1)")
^%ISCLOG("Data",3)=$lb(3,"WorkMgr","CreateWorkers: Creating IPQs","4472","SAMPLES","2016-03-16
17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",3,0)=$lb("baseipq","%ISC_1_","jobs",4,"group","1")
^%ISCLOG("Data",4)=$lb(3,"WorkMgr","CreateWorkers: IPQs complete","4472","SAMPLES","2016-03-16
17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",5)=$lb(3,"WorkMgr","CreateWorkers: Signalled jobs WorkerCmd to 'Add'","4472","SAMPLES","2016-03-16
17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",5,0)=$lb("group","1","jobs",4,"data",$lb("Add","1","%ISC_1_4"))
^%ISCLOG("Data",6)=$lb(2,"WorkMgr","SignalDaemon","4472","SAMPLES","2016-03-16 17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",6,0)=$lb("time",1458148719.334)
^%ISCLOG("Data",7)=$lb(2,"WorkMgr","startWork: Command found","8852","%SYS","2016-03-16 17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",7,0)=$lb("item",$lb("Add","1","%ISC_1_1"),"wait",1)
^%ISCLOG("Data",8)=$lb(3,"WorkMgr","RunDaemon: got signal","8972","%SYS","2016-03-16 17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",8,0)=1
^%ISCLOG("Data",9)=$lb(2,"WorkMgr","startWork: Command found","9184","%SYS","2016-03-16 17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",9,0)=$lb("item",$lb("Add","1","%ISC_1_2"),"wait",1)
^%ISCLOG("Data",10)=$lb(3,"WorkMgr","GroupChange","9184","%SYS","2016-03-16 17:18:39.334","%SYS.WorkQueueMgr","","")
^%ISCLOG("Data",10,0)=$lb("enter",1,"group","1","save",)
```

You'll make breakthroughs

INTERSYSTEMS®

Questions?