12 – Scheduling

1. Process Models

   A. Worker : DBMS component which is responsible for query execution and returning the query results.

   B. Approach

      i. Process per Worker
      relies on OS scheduler,
      using shared memory to maintain global data structure,
      process crash won't kill entire system

      ii. Process Pool
      worker can use any process which is free
      relies on OS and shared memory
      bad for cache locality

      iii. Thread per Worker
      single process can use multiple workers by multiple threads
      DBMS can manage scheduling
      thread crash is critical to system (may kill entire system)

   C. Advantage of multithread architecture
   less overhead of context switch, no shared memory

2. Data Placement

   A. Workers should operate on local data, because of locality problem

   B. Uniform VS. NUMA

      i. Uniform memory access
      memory and core is connected by single system bus
      core has its local cache, but memories are global component.

      ii. Non-Uniform Memory Access
      all core has its local memory and cache
      core is interconnected with each other

C. Memory allocation

i. Until access the memory, OS doesn't allocate memory physically

ii. Approach

1. Interleaving
   distribute allocated memory to all CPUs.

2. First-touch
   allocate memory to CPU's local memory which has thread that accessed the memory location

3. Scheduling

A. Static scheduling

i. DBMS decides number of worker thread to execute query plan when generating query plan

ii. It won't be changed when query is being executed

B. Morsel-driven scheduling

i. Pull based task assignment

ii. Using horizontal partitions of data "morsels" to distribute tasks

iii. Parallel and NUMA-aware operator implementation

C. Actual system architecture

i. Hyper
No dispatcher thread
workers perform cooperative scheduling using a single task queue
if worker doesn't have local task, pull task from global task queue

ii. SAP HANA
maintain soft/hard priority queue
soft queues has tasks that can be steal by other workers
hard queues has tasks that should not be steal by other workers
use "watchdog" thread to check whether groups are saturated and can reassign tasks
NUMA-aware scheduler :
all tasks are in hard queue, since stealing work is not good when there are many number of sockets. Instead, use thread group

iii. SQL server
SQLOS :
user-mode NUMA-aware OS layer that runs in DBMS
manages provisioned hardware