BCM1F TDC

Status of block transfer (BLT) publishing via DIP

Roberval Walsh
DESY

Weekly FCAL/CMS meeting
26.11.2009

BLT_arrays crash

- BLT_arrays (publish BLT via DIP) crashing less than 2 hours running with error:
  terminate called after throwing an instance of 'std::bad_alloc'
  what(): St9bad_alloc
- Still investigating what that means exactly.
- Meanwhile...

BLT_arrays crash debugging

- BLT done without DIP:
  +24h running without crashing (I forced the job to stop);

- BLT done with DIP but only time stamp being published:
  +24h running without crashing (I forced the job to stop);

BLT_arrays crash debugging

- BLT with DIP including time stamp and histograms hits_per_orbit (array size 16384):
  Crashed after 10h running!
- If the hits_in_orbit histograms (array size 115200) are included:
  Crashes in less than 2 hours!

BLT_arrays: DIP client

- DIP browser makes BLT_arrays freeze for minutes while it reads the published histograms.
- A specific DIP client is fast enough reading the published histograms.
- But need to find a way to avoid anyone to use DIP browser for TDC histograms.

BLT_arrays CPU usage

- BLT_arrays uses > 30% of the CPU!
  Not related with DIP!
  To be investigated after solving the crashing problem.

```
top - 17:00:05 up 33 days,  5:40,  3 users,  load average: 0.30, 0.31, 0.27
Tasks:  98 total,   1 running,  97 sleeping,   0 stopped,   0 zombie
Cpu(s):  2.4% us,  6.4% sy,  0.0% ni, 91.0% id,  0.2% wa,  0.0% hi,  0.0% si
Mem:   4144668k total,  4113884k used,    30784k free,    55788k buffers
Swap:  1052248k total,      192k used,  1052056k free,  3868912k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
31184 brmpro    17   0 98.2m  14m 9508 S   35  0.4 549:50.80 BLT_arrays
    1 root      16   0  3532  540  460 S    0  0.0   0:00.89 init
    2 root      RT   0     0    0    0 S    0  0.0   0:01.66 migration/0
    3 root      34  19     0    0    0 S    0  0.0   0:00.09 ksoftirqd/0
    4 root      RT   0     0    0    0 S    0  0.0   0:01.43 migration/1
```

Summary

- Large arrays being published via DIP seems to be the cause of BLT_arrays.
  Under investigation.
- Large arrays also cause the DIP browser to freeze BLT_arrays for long periods of time.
  Investigating solutions.
- BLT_arrays CPU usage is quite large.
  To be investigated.