

Job Instance Statistics Version 4.0

- [Introduction](#)
- [Jobs Statistics File Content](#)
- [Queries](#)
 - [Original 3.1 Query for Job Statistics](#)
 - [All Jobs Statistics \(with the multiplier factor\)](#)

Introduction

Job instance statistics file contains details about a job instance like name, site on which it ran, runtime etc.

Jobs Statistics File Content

Jobs file contains the following information about jobs in the individual workflow.

Job - the name of the job instance

Site - the site where the job instance ran

CondorQTime(sec.) - the time between submission by DAGMan and the remote Grid submission. It is an estimate of the time spent in the condor q on the submit node. The value is calculated as [GRID_SUBMIT/GLOBUS_SUBMIT/EXECUTE -SUBMIT]. The information is obtained from **jobstate** table

Resource(sec.) - the time between the remote Grid submission and start of remote execution. It is an estimate of the time job spent in the remote queue. The value is calculated as [EXECUTE -GRID_SUBMIT/GLOBUS_SUBMIT]. The information is obtained from **jobstate** table

Runtime(sec.) - the time spent on the resource as seen by Condor DAGMan. Is always \geq kickstart. The value is obtained from the *local_duration* in the **job_instance**

Kickstart(sec.) - the actual duration of the job in seconds on the remote compute node. The value is obtained from the *remote_runtime* in the **invocation** table.

Multiplier-Factor - multiplier factor from the user-provided profile that is used to multiply the kickstart time on the remote node. This value is in the **job_instance** table and defaults to 1.

Kickstart_mult(sec.) - the Kickstart time multiplied by the Multiplier-Factor.

Remote-CPU-Time(sec.) - sum of the utime and the stime obtained from the Kickstart invocation record. This value is obtained from the **invocation** table.

Post(sec.) - the postscript time as reported by DAGMan. The value is calculated as [POST_SCRIPT_TERMINATED - POST_SCRIPT_STARTED /JOB_TERMINATED]. The information is obtained from **jobstate** table

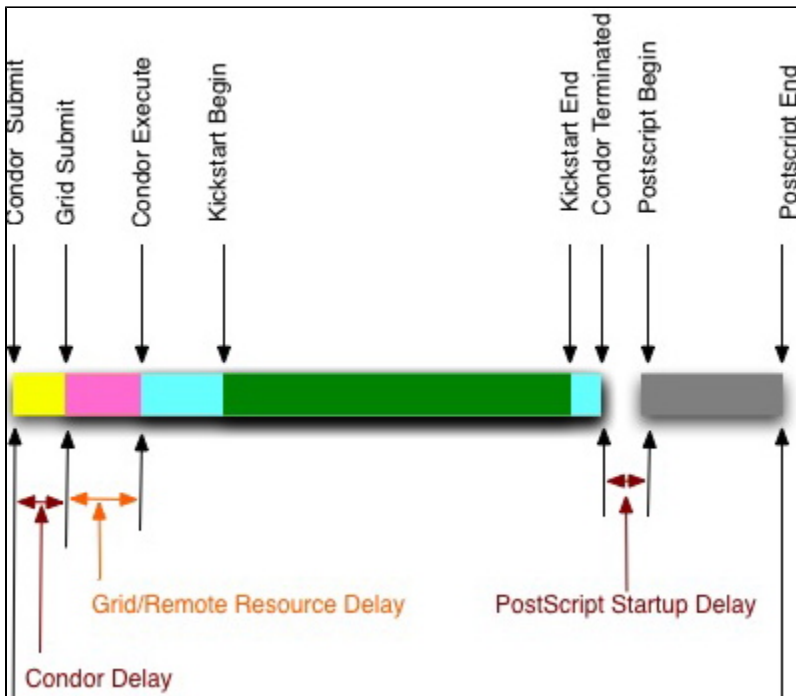
Seqexec(sec.) - the time taken for the completion of a clustered job. This value is obtained from the *cluster_duration* in the **job_instance** table

Seqexec-Delay(sec.) - the time difference between the time for the completion of a clustered job and sum of all the individual tasks kickstart time. This value is obtained as the difference between the *cluster_duration* in the **job_instance** table and sum of all the *corresponding task's remote_runtime* in the **invocation** table.

Exitcode - exitcode from the job. For clustered jobs, it is the highest exitcode found in all the invocation records

Hostname - host name where the job instance ran

Please find below a diagram showing job states and delays.



Queries

The queries for showing information corresponding to jobs in the workflow.

Original 3.1 Query for Job Statistics

```
// API method name: get_job_statistics
select jb.job_id, jb_inst.job_instance_id, jb_inst.job_submit_seq, jb.exec_job_id as job_name, jb_inst.site as
site,
(
(select min(timestamp) FROM jobstate WHERE job_instance_id = jb_inst.job_instance_id and (state =
'GRID_SUBMIT' or state = 'GLOBUS_SUBMIT' or state = 'EXECUTE'))
-
(select timestamp FROM jobstate WHERE job_instance_id = jb_inst.job_instance_id and state = 'SUBMIT' )
) as condor_q_time,
(
(select min(timestamp) FROM jobstate WHERE job_instance_id = jb_inst.job_instance_id and state = 'EXECUTE' )
-
(select timestamp FROM jobstate WHERE job_instance_id = jb_inst.job_instance_id and (state = 'GRID_SUBMIT' or
state = 'GLOBUS_SUBMIT'))
) as resource_delay,
jb_inst.local_duration as runtime,
(
(select sum(remote_duration) FROM invocation as invoc WHERE job_instance_id = jb_inst.job_instance_id and
wf_id = jb.wf_id and task_submit_seq >=0 GROUP BY job_instance_id)
) as kickstart,
(
(select timestamp from jobstate where job_instance_id = jb_inst.job_instance_id and state =
'POST_SCRIPT_TERMINATED')
-
(select max(timestamp) from jobstate where job_instance_id = jb_inst.job_instance_id and (state
='POST_SCRIPT_STARTED' or state = 'JOB_TERMINATED'))
) as post_time,
jb_inst.cluster_duration as seqexec FROM
job as jb, job_instance as jb_inst WHERE
jb_inst.job_id = jb.job_id and
jb.wf_id = 3
ORDER BY jb_inst.job_submit_seq
```

All Jobs Statistics (with the multiplier factor)

```

// API method name: get_job_statistics
select jb.job_id, jb_inst.job_instance_id, jb_inst.job_submit_seq, jb.exec_job_id as job_name, jb_inst.site as
site,
(
(select min(timestamp) FROM jobstate WHERE job_instance_id = jb_inst.job_instance_id and (state =
'GRID_SUBMIT' or state = 'GLOBUS_SUBMIT' or state = 'EXECUTE'))
-
(select timestamp FROM jobstate WHERE job_instance_id = jb_inst.job_instance_id and state = 'SUBMIT')
) as condor_q_time,
(
(select timestamp FROM jobstate where job_instance_id = jb_inst.job_instance_id and state = 'EXECUTE' )
-
(select min(timestamp) FROM jobstate where job_instance_id = jb_inst.job_instance_id and (state='SUBMIT' or
state = 'GRID_SUBMIT' or state = 'GLOBUS_SUBMIT'))
) as resource_delay,
jb_inst.local_duration as runtime,
(
(select sum(remote_duration) FROM invocation as invoc WHERE job_instance_id = jb_inst.job_instance_id and
wf_id = jb.wf_id and task_submit_seq >=0 GROUP BY job_instance_id)
) as kickstart,
(
(select timestamp from jobstate where job_instance_id = jb_inst.job_instance_id and state =
'POST_SCRIPT_TERMINATED')
-
(select max(timestamp) from jobstate where job_instance_id = jb_inst.job_instance_id and (state
='POST_SCRIPT_STARTED' or state = 'JOB_TERMINATED'))
) as post_time,
jb_inst.cluster_duration as seqexec,
(
(select max(exitcode) from invocation as invoc where job_instance_id = jb_inst.job_instance_id and wf_id = jb.
wf_id and task_submit_seq >=0 group by job_instance_id)
) as exit_code,
(
(select h.hostname from host h, job_instance ji where ji.job_instance_id = jb_inst.job_instance_id and h.
host_id = ji.host_id and h.wf_id = 1 GROUP BY ji.job_instance_id)
) as host_name,
multiplier_factor,
(
(select sum(remote_duration * multiplier_factor) FROM invocation as invoc WHERE job_instance_id = jb_inst.
job_instance_id and wf_id = jb.wf_id and task_submit_seq >=0 GROUP BY job_instance_id)
) as kickstart_multi,
(
(select sum(remote_cpu_time) FROM invocation as invoc WHERE job_instance_id = jb_inst.job_instance_id and
wf_id = jb.wf_id and task_submit_seq >=0 GROUP BY job_instance_id)
) as remote_cpu_time

```