Workflow metrics

Overview

This page lists metrics that we wish to derive for Pegasus workflows.

Metrics

Aggregations:

- Total number of workflows executed

```
SELECT count(wf_id) from workflow
```

- Number of workflows executed per day

```
Best aggregated with code glue/datetime libs but currently doable.
```

- Total runtime of each workflow

```
select max(timestamp) - min(timestamp) from jobstate where job_id in (
  select job_id from job where wf_id = (
    select wf_id from workflow where wf_uuid = 'b5310bb2-2871-423d-bdee-8cd0ed1f925a'
  )
)
```

    - Sum of task durations with and without pre/post script

```
select j.job_id,
(select max(timestamp) - min(timestamp) from jobstate where job_id = j.job_id) as total,
(select max(timestamp) - min(timestamp) from jobstate where job_id = j.job_id
  and state not like '%SCRIPT%') as noprepostscript
from job as j
where j.wf_id = (
  select wf_id from workflow where wf_uuid = 'b5310bb2-2871-423d-bdee-8cd0ed1f925a'
)
```

- Total walltime of each workflow
    - Sum of dagman time (difference between dagman end and start times)
- Total number of workflow jobs/tasks executed. This means total jobs and tasks (including failed, repeats, success what not). These are all the job executions for a given workflow.
    - Total number of workflow jobs and tasks that failed
    - Total number of workflow jobs and tasks that succeeded

```
select
count(*) total_jobs,
sum((select count(*) from jobstate where job_id = j.job_id
     and state = 'JOB_SUCCESS')) as job_success,
sum((select count(*) from jobstate where job_id = j.job_id
      and state = 'JOB_FAILURE')) as job_failure,
sum((select count(*) from task where job_id = j.job_id and exitcode = 0)) as task_success,
sum((select count(*) from task where job_id = j.job_id and exitcode <> 0)) as task_failure
from job as j
where j.wf_id = (
  select wf_id from workflow where wf_uuid = 'b5310bb2-2871-423d-bdee-8cd0ed1f925a'
)
```

- Total number of workflow jobs/tasks that were automatically retried.

```
select count(*) from (
  select name, count(job_submit_seq) from job
  where wf_id = (
    select wf_id from workflow where wf_uuid = 'b5310bb2-2871-423d-bdee-8cd0ed1f925a'
  )
  group by name
  having count(job_submit_seq) > 1
)
```

- Workflow jobs/tasks that were retried (breakdown by jobmaname or transformation.

```
select name from job
where wf_id = (
  select wf_id from workflow where wf_uuid = 'b5310bb2-2871-423d-bdee-8cd0ed1f925a'
)
group by name
having count(job_submit_seq) > 1

Etc....
```

- Number of jobs/tasks executed per <time-period>: day, per week , per month, per hour, per year

```
Best aggregated with code glue/datetime libs but currently doable.
```

- Overheads for the jobs (cumulative and average). We should also be able to quantify the % overhead in relation to the overall job time.
  - DAGMan overhead
  - amount of time spent in the Condor Q
  - time from release to the queue to running
  - Kickstart overhead

Related to resource utilization:

- Number of jobs executed on a host/glidein for a particular provisioning request
- Average number of idle jobs in the queue over time (also maybe min/max)
- Average number of running jobs over time (also maybe min/max)
- Average number of idle glide-ins over time (and min/max)

Filters:

- Job type (data transfer in/out, registration, application, other Pegasus jobs)

Graphs:

- Workflows/jobs/tasks over time

Links to Pegasus pages

Gathering Information About a Workflow
Workflow Metrics that can be obtained