SAM-MAIN - Feature #7627

Projects drawing from "live" datasets rather than snapshots

01/12/2015 05:04 PM - Christopher Backhouse

Status:	Rejected	**Start date:**	01/12/2015
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:		**Spent time:**	0.00 hour

Description

This one seems superficially simple, but I realize that it might require a large shake-up of internals. I'd like to understand how feasible it is because it would be very helpful to have.

What I'd like is to be able to create a project where jobs fetch files not from a snapshot of the definition as it existed when the project was submitted, but from a "live" version. If new files are added to the dataset while the project is running, I'd like them to get picked up and processed. An optimization like waiting until the initial snapshot is exhausted before checking for any new files is fine.

Some background on why such a thing would be useful:

In production processing there are several processing tiers a file has to go through. Something like: simulation/daq2raw, reconstruction, PID, CAFing (making DSTs). It would be great to be able to submit multiple of those stages simulataneously, instead of having to wait for all the files to be processed through the step before. In cases where each step is relatively fast (<~ 1 day) I think we spend more time in overhead introduced by this requirement than in actually processing the files.

What we mostly do now is use draining datasets (say, a dataset of reco files that do not have pid children). Initial submission of PID jobs has to wait until a substantial number of reco files are available. This involves whoever's running reco keeping an eye on their jobs, then sending the PID person an email. Often a lot of time can be wasted because one or other of these people is not at their computer.
If the PID is submitted on an incomplete set of reco files, a later submission has to be made to top-up, and submission of this has to wait until all the current PID jobs have finished (and their ouput files made it through FTS) to avoid duplicating the processing of some files.

With "live" datasets you could start both projects at once, and have some kind of automated process check [number of reco files - number of files fetched by PID jobs] and submit PID jobs when necessary, eliminating a lot of dead time.

Another model could be to use recovery datasets. This avoids having to wait for the previous project to finish, and for FTS to do its thing, but it still requires manual construction of multiple datasets (I'd say most processings have each stage submitted three times) with increasingly unweildy definitions. This might be a way this feature could be implemented on the SAM side though.

History

#1 - 01/13/2015 03:07 PM - Robert Illingworth

- Status changed from New to Rejected

The behaviour of snapshots and projects isn't going to change. You can get roughly equivalent functionality by running an initial project on initial_definition, then later running another project using

```
defname: initial_definition minus (snapshot_id x)
```

where x is the snapshot id, which you can get from the project summary. This will exclude all files that are in the original snapshot. You can extend x into a comma separated list x,y,z,... for subsequent jobs. At the end you can run the normal draining type definitions if needed to get any previous files that were missed.

It occurred to me while writing this that a new dimension along the lines of snapshot_by_project_name, which returns all files in the snapshot(s) matching the given project name (which can include wildcards) would make this easier.

#2 - 01/13/2015 05:06 PM - Christopher Backhouse

You'd want it to be OK with there being no matching projects (though that opens you up to typos), so that the first project you submit can be defined the same as all the rest.