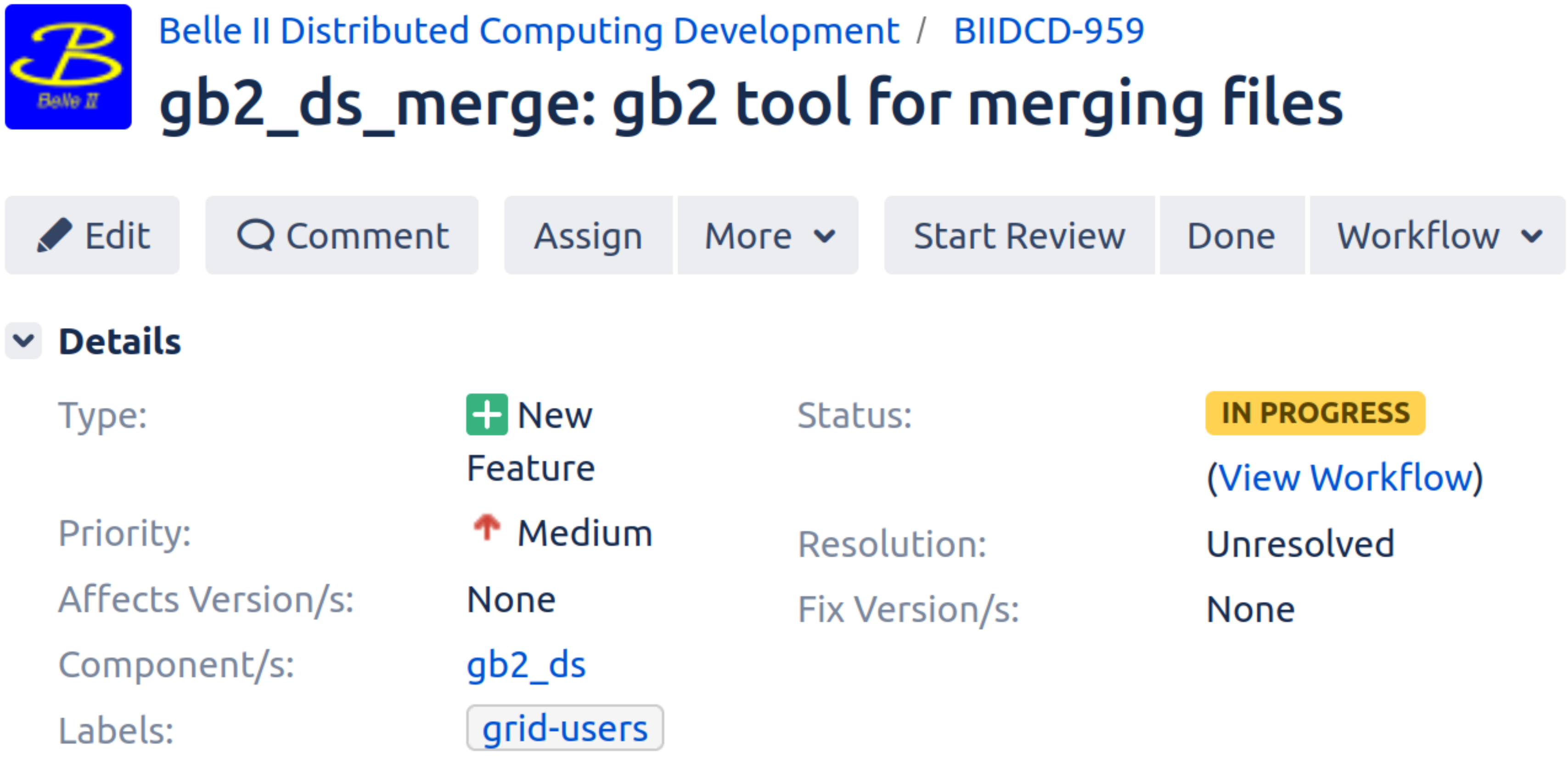
BIIDCD-959



gb2_ds_merge: gb2 tool for merging files

Details

Type:	New Feature	Status:	IN PROGRESS (View Workflow)
Priority:	Medium	Resolution:	Unresolved
Affects Version/s:	None	Fix Version/s:	None
Component/s:	gb2_ds		
Labels:	grid-users		

Jordan Correa (Cinvestav)

Michel Hernández Villanueva (DESY)

Sep 7, 2022

Goal: To implement a new gb2 tool for merging output files. “gb2_ds_merge” (BIIDCD-959).

Method: To take jobs in SEs and to merge them to get less files. These merged ones also be stored in SEs.

Basic functionality: hadd [a_0.root, a_1.root, … a_n.root, b_1.root, b_2.root, …, b_m.root, ...]

>>> [A_0.root, A_2.root, …, A_N.root, B_1.root, B_2.root, …, B_M.root, ...] ; $N < n$, $M < m$.

$$\underbrace{a_0.\text{root},\ a_1.\text{root},\ a_2.\text{root}}_{A_0.\text{root}}\ ,\ \underbrace{a_3.\text{root},\ a_4.\text{root}}_{A_1.\text{root}}\ ,\ \ldots,\ \underbrace{a_{(n-1)}.\text{root},\ a_n.\text{root}}_{A_N.\text{root}}$$

Goal: To implement a new gb2 tool for merging output files. “gb2_ds_merge” (BIIDCD-959).
Method: To take jobs in SEs and to merge them to get less files. These merged ones also be stored in SEs.

Basic functionality: hadd [a_0.root, a_1.root, … a_n.root, b_1.root, b_2.root, …, b_m.root, ...]
>>> [A_0.root, A_2.root, …, A_N.root, B_1.root, B_2.root, …, B_M.root, ...] ; $N < n$, $M < m$.

$$\underbrace{a_0.\text{root},\ \ a_1.\text{root},\ \ a_2.\text{root}}_{A_0.\text{root}}\ ,\ \underbrace{a_3.\text{root},\ \ a_4.\text{root}}_{A_1.\text{root}}\ ,\ \underset{\ldots}{\ldots},\ \underbrace{a_{(n-1)}.\text{root},\ \ a_n.\text{root}}_{A_N.\text{root}}$$

Syntax:

```
gb2_ds_merge  -p <project_name>  -i  <input_file>  --input_lfns <file.txt>  -w <weight>
```

Required params:

- p [PROJECT_NAME] : Name of the output project/folder with the merged jobs.
- i [INPUT_FILE] : Path of input project with/or lfn(s).

Optional_params ("important ones"):

-- input_dslist [FILE] : Uses a file(.txt/.lst, ...) which contains the input lfns to be merged.
- w [WEIGHT] : Maximum size (in MB) of each merged output file.
If not used, a default value will be asigned (let's say 100 MB).
This default value must be greater than the greatest of all input lfns.

Comments (already implemented on local files):

1. A param for the name of the output merged files is not be needed; the tool can identify the name pattern of the input files and to name the output merged files based on that.

   Example:

   ```
   input_lfns = ['<path>/ntuple_chB_ee_xxxxx_jobxxxxx_xx.root',
                 '<path>/ntuple_chB_ee_yyyyy_jobyyyyy_yy.root',
                 '<path>/ntuple_chB_ee_zzzzz_jobzzzzz_zz.root', ...]
   ```

   The name pattern is: ntuple_chB_ee
   The format is: .root

2. Only will be made the hadd (merge) on input files with the same *name pattern* and *format* (.root).
   In future, another formats (.hdf5, ...) will be considered for merging.

3. Each output file will be maked up of *organized* input files. Thus, the order of entries don't be changed.

$$\underbrace{a_0.\text{root},\ \ a_1.\text{root},\ \ a_2.\text{root}}_{A_0.\text{root}}\ ,\ \underbrace{a_3.\text{root},\ \ a_4.\text{root}}_{A_1.\text{root}}\ ,\ \underset{\ldots}{\ldots},\ \underbrace{a_{(n-1)}.\text{root},\ \ a_n.\text{root}}_{A_N.\text{root}}$$

4. The size of each output file will be similar (determined by the *w* param), with exception of the last one (will be lighter).

Questions:

- Where could be better located the new tool *gb2_ds_merge()*, in datasetCLController.py or projectCLController.py?