

Arvados - Feature #13062

[SDK] Reduce collection class memory footprint

02/12/2018 02:57 PM - Peter Amstutz

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assigned To:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	To Be Groomed		
Description			
Reduce collection class memory footprint in order to reduce the footprint of arv-mount and arvados-cwl-runner in order to run on smaller, cheaper nodes.			
General approach: instead of parsing the manifest once and creating Python objects for every directory and file, reparse and create python objects on demand.			
Possibly strategy:			
<ul style="list-style-type: none">• Initial manifest parsing creates an index that maps each directory path to one or more manifest streams (by offset or by using memoryview) which describe the contents of that directory.• When the contents of a Collection or Subcollection is needed, look up the stream(s) associated with the Directory from the index and parse them.• Consider doing something similar at individual file level, only load "segments" on demand (may come at cost of higher overhead if it turns out the client is going to visit most of the files in a given directory anyway).• Make it possible for a caching strategy to evict loaded collection contents / file segments.			
Challenges:			
<ul style="list-style-type: none">• Can't cache evict anything that's been returned to the (Python SDK) user unless we can determine it isn't being held (maybe requires reference counting scheme).• For FUSE, stable inode assignment like https://dev.arvados.org/issues/12664#note-2 would allow us to evict things from the FUSE cache that are still known to the kernel.			

History

#1 - 02/12/2018 02:57 PM - Peter Amstutz

- Status changed from New to In Progress

#2 - 02/12/2018 03:31 PM - Peter Amstutz

- Description updated

- Status changed from In Progress to New