Ceph - Feature #609

osd: query pool/pg for objects with given xattr

11/23/2010 11:42 AM - Sage Weil

Status:	Resolved	**% Done:**	0%
Priority:	Normal	**Spent time:**	0.00 hour
Assignee:	Yehuda Sadeh		
Category:	OSD		
Target version:	v0.25		
Source:		**Reviewed:**	
Tags:		**Affected Versions:**	
Backport:		**Pull request ID:**	

Description

This will probably take the form of a pool class plugin?
It could start as just a hack, for now.

History

#1 - 11/29/2010 10:18 AM - Sage Weil

- Target version changed from 12 to v0.25

#2 - 11/30/2010 03:56 PM - Sage Weil

- translation missing: en.field_position set to 3

#3 - 12/11/2010 01:54 PM - Sage Weil

- Assignee set to Yehuda Sadeh

#4 - 12/12/2010 09:43 PM - Sage Weil

- Subject changed from osd; query pool/pg for objects with given xattr to osd: query pool/pg for objects with given xattr

#5 - 01/03/2011 05:27 PM - Yehuda Sadeh

The problem that I see is that this is going to be a very slow operation, and we need to figure out how not to timeout. One solution is that we return immediately with an 'operation token', and later on we can poll the osd with this token, so that we get the results that were accumulated in the mean time.
Another easier option, is just have it similar to a read-first - read-next sequence. The client first sends an initial request: pgcall-init, then the client sends pgcall-process(x) where x is the number of entries the client wants to go over. The osd then processes that amount of entries and only responds with the entries that weren't filtered out (can be an empty response). Then the client iterates over until everything is completed. This will be slower than the first solution, however, much simpler.

#6 - 01/03/2011 07:52 PM - Sage Weil

Can we use the list objects operation (LSPG?) as a model here? That's basically what it does.. it grabs up to N items and there's a cookie to continue the operation.

#7 - 01/04/2011 02:47 PM - Yehuda Sadeh

- Status changed from New to Resolved

implemented it as a new PGLS_FILTER operation, but reuses the PGLS operation code. There's no class extension now, but in the future we might

want to add that (once we have some more use cases).
The problem with creating new pg class methods is that they'll run in a different context than the current object class method, hence they'll require a different api. We think that at this point where we only have a single case we'd be better off just implementing the filtering option as a new op. In any case, we're passing enough parameters to the new op so that if we decide to implement a pg class mechanism, it'd be compatible.

Implemented in 1b0bb7a1e95da0de29adbbc57c2b299a66a19ee5

#8 - 01/05/2011 03:14 PM - Yehuda Sadeh

- Status changed from Resolved to In Progress

#9 - 01/06/2011 03:02 PM - Yehuda Sadeh

```
yehudasa@skinny:~/ceph/src$ ./rados ls -p metadata --filter="parent 0x100"
602.00000000
dirino=100 dname=stray2 v=1
604.00000000
dirino=100 dname=stray6 v=1
605.00000000
dirino=100 dname=stray7 v=1
606.00000000
dirino=100 dname=stray3 v=1
607.00000000
dirino=100 dname=stray1 v=1
603.00000000
dirino=100 dname=stray0 v=1
```

#10 - 01/19/2011 09:15 PM - Sage Weil

- Status changed from In Progress to 7

#11 - 03/01/2011 04:38 PM - Sage Weil

- Status changed from 7 to Resolved