rgw - Bug #5810

rgw: update_type does not work correctly when meta object does not exist (or removed)

07/30/2013 05:55 PM - Yehuda Sadeh

Status:	Rejected	**% Done:**	0%
Priority:	Urgent	**Spent time:**	0.00 hour
Assignee:	Yehuda Sadeh		
Category:			
Target version:	v0.67 - Dumpling		
Source:	other	**Affected Versions:**	
Tags:		**ceph-qa-suite:**	
Backport:		**Pull request ID:**	
Regression:	No	**Crash signature (v1):**	
Severity:	3 - minor	**Crash signature (v2):**	
Reviewed:			

Description

Two related but separate issues:
- when creating a new object, update-type=update-by-version doesn't work at all (there's no older version)
- object removal does not look at versions at all

History

#1 - 07/30/2013 05:55 PM - Yehuda Sadeh

- Priority changed from Normal to Urgent

#2 - 07/30/2013 09:55 PM - Ian Colle

- Assignee set to Yehuda Sadeh

#3 - 07/31/2013 10:08 AM - Yehuda Sadeh

- Target version set to v0.67 - Dumpling

#4 - 07/31/2013 11:11 AM - Yehuda Sadeh

Issue #2 (not checking versions for removal operations) can be fixed easily.

Was thinking about issue #2. Note that it only manifests itself if the target metadata object does not exist. So if it exists but has newer version (even on a newer instance) we don't have a problem, as when we do the partial sync we match tags when applying.
Here we also have two different issue. The easy problem is that we try to match tag and it doesn't work as the target object does not exist. We can fix that by not matching tags when the target object does not exist. However, we have the issue of when doing so we'd end up with reverting to older version (and then moving forward again) when the request originated at the current zone. For which I think the correct solution is to keep track on the source region:zone where the operation originated and never apply it there.

#5 - 07/31/2013 12:00 PM - Yehuda Sadeh

- Status changed from New to Rejected

Ok, after discussing that with Josh, he reminded me that we don't really replay the log. What we do is always fetching the latest from the master. In this case, both issues are moot.