openQA Infrastructure - action #57494

increase space for o3, potentially split /assets and /results same as for osd

2019-09-29 10:22 - okurz

Status:	Resolved	**Start date:**	2019-09-29
Priority:	High	**Due date:**	2020-04-12
Assignee:	okurz	**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:	Current Sprint		

Description

Motivation

We currently have 4TB for o3 but we recently struggle again to accomodate all necessary data. Some job groups are hardly able to sustain even a single repo snapshot of Factory, see #57338 . For OSD we recently switched to have separate partitions for /assets and /results , probably we should do the same for o3.

History

#1 - 2019-09-29 10:26 - okurz

- Status changed from In Progress to Feedback

https://infra.nue.suse.com/SelfService/Display.html?id=146940

#2 - 2019-10-01 09:00 - okurz

- Status changed from Feedback to Blocked

#3 - 2019-10-29 11:05 - michel_mno

oups wrong issue

#4 - 2019-12-18 11:03 - okurz

- Status changed from Blocked to Feedback

- Priority changed from Normal to High

- Target version set to Current Sprint

hmpf, ticket in "openqa" queue was resolved without any further comment. Reopened as https://infra.nue.suse.com/SelfService/Display.html?id=158336

But further it seems we have again a problem right now with too many results. Leap 15 shows long result+log retention periods: we have currently "important logs" 365d, "important results" 3650d

The SQL query select group_id,count(id) from jobs where logs_present=TRUE group by group_id order by count; shows

```
 group_id | count
----------+-------
[…]
       50 |  6720
```

We should ask lnussel, maxlin, lkocman if we can reduce the log retention for leap. coolo and me did so on [#opensuse-factory](irc://chat.freenode.net/opensuse-factory) :

```
18/12/2019 11:58:40] <coolo> maxlin, DimStar: how far back do you need openqa logs on o3?
[18/12/2019 11:59:56] <DimStar> coolo: you mean on the tests? or logs like syncd?
[18/12/2019 12:00:36] <coolo> test results
[18/12/2019 12:01:51] <maxlin> do we ran out of space on o3?
[18/12/2019 12:02:07] <coolo> if not this week, then next
[18/12/2019 12:02:27] <okurz> maxlin, lnussel: Who is the main person for Leap on o3? I guess lkocman but coul
dn't find him here. Regarding to what coolo asks. Currently the main "Leap" job group on o3 has very long rete
ntion periods, e.g. 365d for "important" logs and "3650d" for "important results" and we currently do not have
 enough space to accomodate that much. either we reduce it or we find more disk space really fast which will o
nly happen with
[18/12/2019 12:02:27] <okurz> a good escalation
```

Further:

```
[18/12/2019 12:07:40] <lnussel> yast logs and videos for not linked jobs don't have to stay long. either we re
view something in time or never
[18/12/2019 12:07:53] <coolo> but it's important to differentiate your needs for test results and for logs
[18/12/2019 12:07:56] <okurz> define "not long" and "in time"?
[18/12/2019 12:08:17] <coolo> lnussel: a week? two?
[18/12/2019 12:08:26] <lnussel> I'd have guesses something like that too
[18/12/2019 12:08:40] <maxlin> a week should good enough IMO
[18/12/2019 12:08:49] <coolo> let's make it 10 days then?
[18/12/2019 12:08:53] <lnussel> k
[18/12/2019 12:08:57] <coolo> (right now we have set 60, which is insane)
[18/12/2019 12:09:11] <lnussel> still requesting more disk space makes sense as this issue will come again
[18/12/2019 12:09:27] <coolo> ok, now that we discussed logs - how long do you need results of daily builds?
[18/12/2019 12:09:33] <coolo> lnussel: go ahead!
[18/12/2019 12:11:13] <coolo> are 2 months for results of unlabeled builds good?
[18/12/2019 12:11:41] <lnussel> I hope so, yes
[18/12/2019 12:11:55] <lnussel> how much space are we talking about per build here?
[18/12/2019 12:13:04] <coolo> lnussel: one job is 50MB on average. one build has 130 jobs - clones not counted
[18/12/2019 12:13:18] <maxlin> the oldest bug I've had tagged on current snapshot is https://bugzilla.suse.com
/show_bug.cgi?id=1158873
[18/12/2019 12:13:29] <maxlin> 2 months should be ok
[18/12/2019 12:13:40] <|Anna|> SUSE bug 1158873 in openSUSE Distribution "[Build 536.2] openQA test fails in r
andom places on GNOME - wayland session suddenly crashed(sporadic)" [Normal, New]
[18/12/2019 12:13:47] <okurz> "oldest bugs" are not relevant. it's "oldest job" on o3
[18/12/2019 12:14:06] <coolo> good. I harmonized TW and Leap now to keep tagged builds forever, untaged logs f
or 10 days, tagged logs for 120 days and untagged results for 80 days
[18/12/2019 12:14:35] <okurz> lnussel: I doubt we will get "more space" without sponsoring and management esca
lation stating the business importance
```

so coolo set to "Keep logs for" 10, "Keep important logs for" 120, "Keep results for" 80, "Keep important results for" 0. Currently /space is on 93%, 304G free. over the past weeks /assets was around 89-90%. https://nagios-devel.suse.de/pnp4nagios//index.php/graph?host=ariel-opensuse.suse.de&srv=space_partition&start=1576235168&end=1576649843 shows the recent rise yesterday morning. Not sure if it's "just" more Leap 15 jobs now but of course critical bugs in parents cancelling whole clusters, especially kernel tests, could help us to save space.

#5 - 2020-01-10 12:00 - okurz

wrote on https://trello.com/c/JQtnALhz/6-openqa-hw-budget-planning#comment-5e185a3e9a5c3786c32fd089 for the kind request to have budget for an increase of available space.

#6 - 2020-02-21 13:29 - okurz

no response, no actions in neither trello nor the infra-ticket.

#7 - 2020-03-02 21:36 - okurz

- Status changed from Feedback to Blocked

Created new ticket https://infra.nue.suse.com/SelfService/Display.html?id=164966 as the old one is in the "openqa" queue which we can't change and is kinda stale.

#8 - 2020-04-03 20:15 - okurz

- Due date set to 2020-04-12

- Status changed from Blocked to In Progress

SUSE IT was very nice to us and we have more space now:

```
# lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
vda    254:0    0    10G  0 disk
├─vda1 254:1    0   9.8G  0 part /
└─vda2 254:2    0 250.7M  0 part [SWAP]
vdb    254:16   0     5T  0 disk
└─vdb1 254:17   0     4T  0 part /space
vdc    254:32   0   100G  0 disk /var/lib/pgsql
vdd    254:48   0     5T  0 disk
```

however either this caused #65202 or at least we might want to wait for #65202 before we extend the existing filesystem. On OSD we use

```
$ ssh osd lsblk
NAME   MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
loop0    7:0    0     5G  0 loop /home
```

```
vda    253:0    0    10G  0 disk
├─vda1 253:1    0   9.8G  0 part /
└─vda2 253:2    0 250.7M  0 part [SWAP]
vdc    253:32   0    80G  0 disk /srv
vdd    253:48   0     5T  0 disk /results/share
vde    253:64   0     5T  0 disk /var/lib/openqa
```

with a mount setup:

```
/results /var/lib/openqa none bind 0 0
/dev/vdd /assets xfs defaults,logbsize=256k,noatime,nodiratime 1 2
/assets /var/lib/openqa/share none bind 0 0
/dev/vde /results xfs defaults,logbsize=256k,noatime,nodiratime 1 2
```

with o3 mount points:

```
UUID="b78ac13f-2c3a-49ed-8db1-36d0f1bafd98" /space xfs defaults,logbufs=8,logbsize=256k  1 2
/space/openqa /var/lib/openqa bind bind 0 0
/space/snapshot-changes /var/lib/snapshot-changes bind bind 0 0
```

I wonder if we should use noatime,nodiratime on o3 as well.

We should be able to create a new mount point /assets for vdd, move all content from /var/lib/openqa/share there and bind mount same as on osd.

Running benchmark on near-idle system:

```
zypper -n in fio
cd /space
systemctl stop apache2 rsyncd openqa-webui openqa-scheduler && fio --name TEST --eta-newline=5s --filename=fio
-tempfile.dat --rw=randrw --size=500m --io_size=10g --blocksize=4k --ioengine=libaio --fsync=1 --iodepth=1 --d
irect=1 --numjobs=1 --runtime=60 --group_reporting && systemctl start apache2 rsyncd openqa-webui openqa-sched
uler
```

result:

```
…
Run status group 0 (all jobs):
   READ: bw=171KiB/s (175kB/s), 171KiB/s-171KiB/s (175kB/s-175kB/s), io=10.0MiB (10.5MB), run=60009-60009msec
  WRITE: bw=179KiB/s (183kB/s), 179KiB/s-179KiB/s (183kB/s-183kB/s), io=10.5MiB (11.0MB), run=60009-60009msec

Disk stats (read/write):
  vdb: ios=2752/9887, merge=0/142, ticks=30048/64268, in_queue=95028, util=70.01%
```

and for the new device /dev/vdd

```
mkfs.xfs /dev/vdd
mount -o logbufs=8,logbsize=256k,noatime,nodiratime /dev/vdd /mnt
fio --name TEST --eta-newline=5s --filename=fio-tempfile.dat --rw=randrw --size=500m --io_size=10g --blocksize
=4k --ioengine=libaio --fsync=1 --iodepth=1 --direct=1 --numjobs=1 --runtime=60 --group_reporting
                                                    ```
```

```
result:
```

…
Run status group 0 (all jobs):
READ: bw=191KiB/s (195kB/s), 191KiB/s-191KiB/s (195kB/s-195kB/s), io=11.2MiB (11.7MB), run=60025-60025msec
WRITE: bw=197KiB/s (202kB/s), 197KiB/s-197KiB/s (202kB/s-202kB/s), io=11.6MiB (12.1MB), run=60025-60025msec

Disk stats (read/write):
vdd: ios=2861/8780, merge=0/4, ticks=38872/5172, in_queue=51104, util=85.08%

```
so slightly higher numbers but not significant to consider /dev/vdd the faster one. So I will decide for using
 /dev/vdd as the new target for "/assets". To decide in a very simple test if LVM has an unexpected performanc
e impact I will conduct a benchmark after creating a logical volume from which I can create a filesystem:
```

pvcreate /dev/vdd
vgcreate vg0 /dev/vdd
lvcreate -L 3T -n assets /dev/vg0
mkfs.xfs /dev/vg0/assets
mount -o logbufs=8,logbsize=256k,noatime,nodiratime /dev/vg0/assets /mnt
cd /mnt/
fio --name TEST --eta-newline=5s --filename=fio-tempfile.dat --rw=randrw --size=500m --io_size=10g --blocksize=4k --ioengine=libaio --fsync=1
--iodepth=1 --direct=1 --numjobs=1 --runtime=60 --group_reporting

```
results:
```
```

Run status group 0 (all jobs):
READ: bw=170KiB/s (174kB/s), 170KiB/s-170KiB/s (174kB/s-174kB/s), io=9.96MiB (10.4MB), run=60001-60001msec
WRITE: bw=177KiB/s (182kB/s), 177KiB/s-177KiB/s (182kB/s-182kB/s), io=10.4MiB (10.9MB), run=60001-60001msec

Disk stats (read/write):
dm-0: ios=2547/10186, merge=0/0, ticks=42136/15436, in_queue=57572, util=95.93%, aggrios=2557/10229, aggrmerge=0/4, aggrticks=37360/6840, aggrin_queue=49704, aggrutil=82.56%
vdd: ios=2557/10229, merge=0/4, ticks=37360/6840, in_queue=49704, util=82.56%

rerun:

Run status group 0 (all jobs):
READ: bw=392KiB/s (402kB/s), 392KiB/s-392KiB/s (402kB/s-402kB/s), io=22.0MiB (24.1MB), run=60005-60005msec
WRITE: bw=396KiB/s (406kB/s), 396KiB/s-396KiB/s (406kB/s-406kB/s), io=23.2MiB (24.4MB), run=60005-60005msec

Disk stats (read/write):
dm-0: ios=5880/22524, merge=0/0, ticks=40148/17884, in_queue=58036, util=96.82%, aggrios=5884/22551, aggrmerge=0/0, aggrticks=31108/2296, aggrin_queue=48488, aggrutil=80.64%
vdd: ios=5884/22551, merge=0/0, ticks=31108/2296, in_queue=48488, util=80.64%

so first a slight decrease but then double the value? Not reliable to me. Anyway, I can keep LVM :)

Added entries to /etc/fstab:

/dev/vg0/assets /assets xfs defaults,logbufs=8,logbsize=256k,noatime,nodiratime 1 2

okurz: 2020-04-03: Enable the following after having synced all assets, see https://progress.opensuse.org/issues/57494

#/assets /var/lib/openqa/share bind bind 0 0

and now

mount -a
rsync --delete -avHP /var/lib/openqa/share/factory/hdd/fixed/ /assets/factory/hdd/fixed/ && rsync --delete -avHP /var/lib/openqa/share/factory/iso/fixed/ /assets/factory/iso/fixed/ && rsync --delete -avHP /var/lib/openqa/share/ /assets/

to first sync the fixed assets which are changing less often.
This will run for some time. Then I can do it again repeatedly until the diff is small, then switch off services, move over the rest, switch mount points and re-enable services:

rsync --delete -avHP /var/lib/openqa/share/ /assets/ && sed -i 's@#/assets@/assets@' /etc/fstab && systemctl stop apache2 rsyncd openqa-webui openqa-scheduler openqa-gru openqa-websockets && mv /var/lib/openqa/share{,.old}/ && mkdir /var/lib/openqa/share/ && mount -a && systemctl start apache2 rsyncd openqa-webui openqa-scheduler openqa-gru openqa-websockets && rsync -avHP /var/lib/openqa/share.old/ /assets/

or something like that :)

EDIT: 2020-04-04: While syncing is going on I can also resize the existing partition and filesystem for results:

parted -s /dev/vdb resizepart 1 100%
xfs_growfs /space/

Bumped some retention limits and asset quotas for the groups that were barely holding single builds

EDIT: 2020-04-05: The sync completed. The above command for sync and switch-over worked fine.

#9 - 2020-04-05 18:38 - okurz

- Status changed from In Progress to Resolved

I deleted all old data from /space. Currently we have

```
# df -h | grep '\(space\|assets\)'
/dev/vdb1               5.0T  2.3T  2.8T  45% /space
/dev/mapper/vg0-assets  3.0T  1.6T  1.5T  51% /assets
```

so enough of headroom :)

I have bumped further some quotas and log retention periods walking over all job groups. I guess for now there is nothing more to do here.

Talked with gschlotter. Both the old and new storage is on rotating disk so less performant than SSD but cheaper. So eventually we might have the possibility to use SSD based storage.