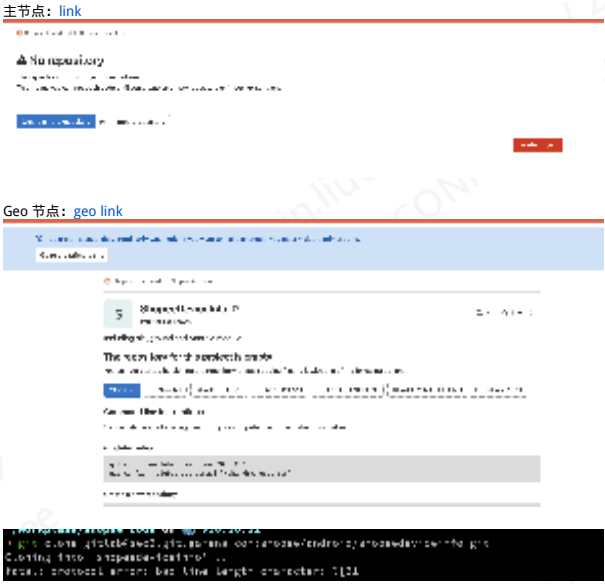



git check 脚本结果问题分类

问题分类	现象	迁移后影响面	相关库ID、Name	是否需要修复 / 修复方案	修复结果	Root Cause
情况1: 主节点中数据库有数据，裸库没有磁盘文件 Geo节点数据库有数据，裸库有磁盘文件	<p>主节点: link</p>  <p>Geo 节点: geo link</p>  <pre>Warning: Repository code on /dev/zero git clone git@github.com:shopee/shopee.git Cloning into 'shopee'... fatal: repository has line length protection: 131</pre>	<ul style="list-style-type: none">以前无法使用，页面显示No Repository，无法点击进入这个库并使用相关的功能，迁移后相当于创建了一个新的同名代码仓库，可以点击进入该代码库，也可以使用相关的功能，也可以clone此代码库（空库，无ref）目前创建代码是使用space ticket自动创建，因为迁移前存在数据库数据，没有裸库数据，迁移后数据库数据和裸库数据同时存在，则可能用户在再次创建与此代码库同名的ticket时，ticket执行实际代码库创建动作时报错（因为裸库已经存在了）	1. 12625-ShopeeDevvicelinfo	待定/如果需要修复，可以在跟用户确认后从页面上将库进行删除	如果有需要，10k节点升主后处理，影响面可控	GEO同步本身会根据praefect路由在相应的gitaly集群上创建代码库及路由信息，然后利用git fetch进行git refs/object的同步。所以如果数据库里有数据，当10K新集群挂为2K老集群的GEO节点进行同步时，由于10K集群上没有这个代码库，会在新的10K集群上创建一个同名的裸库。但是原集群是没有这个裸库的，所以迁移前后则会有数据不一致的问题。

8. 19
29
7 -
m
ob
ile-
we
b-
no
ke
ys
9. 19
32
5 -
sh
op
ee-
url
-
lib
10. 23
75
0 -
ro
ut
er
11. 57
47
0 -
sel
ler
da
ta-
air
flo
w
(主
节
点
操
库
被
删
除)
12. 76
09
6 -
be
es
ho
p
htt
ps:
//g
it.
ga
re
na.
co
m
/xi
ng
zhi
-
yo
u
/b
ee
sh
op
13. 76
67
2 -
sh
ar
ed
-
sto
ck-
ser
vice
14. 76
64
6 -
lp
Se
rvi
ce

15. 76
12
0 -
idf
dc
on
lin
e
m
od
elt
est
16. 75
69
8 -
Da
tas
tre
a
m
Es
(主
节
点
裸
库
被
删
除)
17. 75
69
0 -
Fo
od
Co
m
m
on
18. 75
29
1 -
Sq
eB
atc
hI
m
po
rt
19. 74
99
4 -
De
ep
Ma
tch
20. 74
85
8 -
Go
qu
ery
21. 74
51
0 -
Ex
a
m
ple
22. 74
32
2 -
V
M
Co
ntr
oll
er
23. 72
99
3 -
Ch
at
bo
tCi
cd
24. 72
95
8 -
Go
Mi
cro
25. 72
66
8 -
en
try

26. 72
32
2 -
log
o-
de
tect
27. 71
75
2 -
DR
Co
ntr
oll
er
28. 71
08
0 -
Alg
ori
th
mF
ro
nt
end
29. 68
38
3 -
de
co
rat
io
n-
sty
le-
te
m
pla
tes
30. 67
63
3 -
Ele
m
en
tUi
31. 67
59
7 -
da
ta_
ser
vice
32. 67
56
7 -
Da
taS
tu
di
oT
est
inga
33. 67
01
6 -
ra
p-
pl
ugi
ns
34. 66
89
6 -
Un
ifie
dLi
nk
35. 66
73
8 -
Im
scr
ipts
36. 66
25
6 -
La
m
bda

37. 66
04
9 -
Ite
m
Ma
rtR
eal
ti
m
eU
til
38. 65
40
9 -
sh
op
39. 64
67
6 -
Es
Ad
min
40. 64
32
3 -
Cv
Alg
ori
th
m
Sd
kR
ele
ase
41. 64
26
8 -
m
od
elb
an
k-
ap
pli
cat
ion
42. 63
96
4 -
Ch
ro
m
eE
xte
nsi
on
Sg
43. 63
46
8 -
Pr
ot
oc
Ge
nG
oC
ha
ssi
sG
rpc
44. 63
28
5 -
rat
ing
-
rn-
lib
45. 62
68
9 -
Sh
op
ee
Co
re
46. 62
64
3 -
lo
wC
od
e-
fe

47. 61
98
5 -
M
M
CE
nd
po
int
48. 61
23
8 -
Ma
ter
ial
nt
er
na
tio
nal
iza
tio
n
l
os
49. 61
23
6 -
Ma
ter
ial
Co
m
po
ne
nts
los
50. 60
32
3 -
lan
di
ng
pa
ge-
bff
51. 60
32
2 -
M
BP
-
ad
min
52. 60
30
7 -
Dis
co
ve
ry
ad
sA
de
ngi
ne
Co
m
m
on
53. 60
18
6 -
Ap
iLi
stA
ut
o
m
at
or
54. 59
77
4 -
pri
vate
55. 59
27
9 -
ssf
56. 58
56
2 -
M
m
uS
er
ver

57. 58
07
7 -
Da
ta
Qu
ality
58. 57
91
5 -
BIT
oo
lsT
est
ing
59. 57
69
8 -
tsc
p
ms
60. 55
89
8 -
go
_se
lle
r_lib
61. 55
16
9 -
Co
ve
r1
m
gV
2.2
.0
62. 54
92
5 -
m
an
sib
le
63. 54
48
0 -
Pr
od
uct
La
bel
Da
ta
Mi
gr
ati
on
64. 54
04
4 -
S
ms
ser
vice
65. 53
80
7 -
lite
bff
66. 53
68
8 -
sc
or
m-
m
ock
67. 53
07
2 -
Biz
plt
we
bE
ntr
yta
sk
68. 52
80
1 -
ra
nk
_pr
oxy

69. 52
43
9 -
S3
Kit
Sa
m
ple
Us
ag
ec
opy
70. 51
97
1 -
an
-
pu
pp
ete
er
71. 51
94
0 -
Se
ata
lk5
pe
csl
ibr
ar
y2
72. 51
86
7 -
liu
-
kun
73. 51
85
2 -
Pu
pp
ete
er
74. 51
12
8 -
m
on
ke
y-
m
ast
er
75. 51
71
8 -
Un
ifo
rm
Pr
edi
ctor
76. 49
60
3 -
ch
as
sis
-
te
m
pla
te
77. 49
02
1 -
toc
re
po
rt
78. 49
01
5 -
res
ou
rce
_m
an
ag
e_
be
79. 48
57
1 -
dp
ap
-fe

80. 48
10
1 -
In
Ho
us
eF
ac
eD
ete
ctor
81. 46
58
7 -
sh
op
ee
_lo
g_s
er
ve
r_c
lo
ne
82. 46
35
8 -
De
m
oC
opy
83. 44
87
5 -
Na
m
En
try
Ta
sk
84. 44
55
1 -
go
-
ch
as
sis
-
w
ms
85. 44
36
9 -
Op
sS
ha
re
Vu
eE
nv
86. 42
68
7 -
mk
t-
gin
87. 42
54
0 -
Lo
gA
nal
ysis
88. 42
37
8 -
Py
sp
ar
kD
ata
Pi
pel
ine
89. 40
64
4 -
L4
Rel
ea
se
90. 40
50
1 -
tes
t3

91. 40
38
5 -
se
arc
hp
lt-
co
m
m
en
td
ata
92. 35
45
0 -
Se
ata
lkA
pp
Cli
93. 35
01
4 -
ad
s_s
tat
isti
cs
94. 34
95
7 -
t5-
mi
rror
95. 34
93
8 -
yul
eo
n_t
est4
96. 34
93
7 -
t2-
mi
rror
97. 33
51
8 -
mt
rai
ni
ng
98. 33
43
5 -
ngi
nx-
au
to
99. 80
58
-
db
-
ing
est
io
n-
to
olt
est
100. 99
68
-
ss
pp
ay
m
en
tkit
101. 99
62
-
SS
PP
ay
m
en
tKit

102. 10
16
2 -
cre
dit
_se
rvi
ce_
de
mo
103. 49
23
-
sh
op
ee-
so
up
-
to
ols
104. 94
16
-
de
vo
ps
105. 43
75
-
my
sql
_pr
oxy
106. 35
99
-
my
sql
_pr
oxy
107. 30
90
3 -
Ne
wj
oi
ne
rTr
ain
ing
Ta
sk
108. 30
10
8 -
Lis
tSe
rvi
ce2
109. 30
10
7 -
Lis
tSe
rvi
ce
110. 29
89
7 -
ge
o-
wa
re
ho
use
111. 26
92
0 -
Pr
oje
ct
Ha
wk
eye
112. 25
78
7 -
M
ult
im
edi
aS
er
ve
rIn
ter
face

			113. 25 20 0 - An dr oi dff m pe gT ool			
情况2: HEAD 指向错误	<p>主节点:</p> <p>项目1</p> <p>报错:</p> <pre> git clone https://github.com:weibin.liu@shopee.com:InteractionsBazaar.git Cloning into 'InteractionsBazaar'... remote: Counting objects: 1022, done. remote: Counting objects: 1022 (1022/1022), done. remote: Compressing objects: 100% (1022/1022), done. remote: Total 1022 (delta 622), reused 1022 (delta 0), pack reused 0 Receiving objects: 100% (1022/1022), 251.22 KiB 1.26 MiB/s, done. Resolving deltas: 100% (1022/1022), done. warning: remote HEAD refers to nonexistent ref, unable to checkout. </pre> <p>项目2:</p> <p>报错:</p> <pre> git clone https://github.com:weibin.liu@shopee.com:InteractionsBazaar.git Cloning into 'InteractionsBazaar'... remote: Counting objects: 169, done. remote: Counting objects: 169 (169/169), done. remote: Compressing objects: 100% (169/169), done. remote: Total 169 (delta 88), reused 169 (delta 88), pack reused 0 Receiving objects: 100% (169/169), 34.77 KiB 685.88 KiB/s, done. Resolving deltas: 100% (88/88), done. WARNING: remote HEAD refers to nonexistent ref, unable to checkout. </pre>	<p>迁移前后都会报错（前后一致），报错影响：git clone时可以拉到全部的refs和objects，但是无法自动checkout到HEAD，有一个warning，需要用户手动checkout到想使用的分支（并不影响代码库refs和objects的数据完整性）</p>	<p>1. 24 04 - Int ers ect io nO bs er ve r 无 法 直 接 搜 到 已 被 归 档 https://github.com/shopee/InteractionsBazaar</p> <p>2. 68 0 - Da ta Bi nd ing De mo</p> <p>3. 64 6 - ad mi n. re de e m. th un de rst rik e. ga re na. com</p>	是/修改HEAD指向 云辉建议修复此类代码库	已完成	裸库中HEAD指针指向的分支被删除了，导致git clone后自动checkout HEAD时找不到HEAD指针指向的分支。 例如HEAD指向了refs/heads/master，但是裸库内并无master分支
情况3: 主节点裸库有invalidate object (数据库中有project数据，裸库内有一个invalidate object对象)	<pre> git fetch https://github.com:weibin.liu@shopee.com:InteractionsBazaar.git git: lock: write object file: 80c4e744d17f43109c...80c4e744d17f43109c </pre>	迁移前后一致，都能进入代码库（空库），迁移后invalidate的objects会消失	1. 10 47 4 - pi pel ine tes tsh3	无需修复/迁移后无影响	无需处理	造成此类原因的case较多，猜测是裸库删除后又重新建了新的同名库，此类情况这里不做深究

2. 62
93
0 -
pr
oa
cti
ve
api
3. 63
01
3 -
sp
ac
e-
we
nw
en
-
we
nw
en2
4. 63
07
4 -
sh
op
ee-
co
re
da
ta
(in
de
x
/p
ac
k
记
录
为
空
,
可
以
认
为
是
空
库)
5. 64
08
4 -
po
ds
pec
6. 65
96
2 -
en
try
-
task
7. 66
69
0 -
ivp
lat
for
m
m
od
el
8. 67
36
3 -
Ins
tat
io
n-
ASM

GEO同步时可能会失败（因为GEO同步依赖git fetch），但是由于同步时会创建新的裸库，且迁移前原裸库是带有invalidate objects的空库（invalidate object由于悬空所以正常clone/pull时不会被拉下来，所以不会受影响），迁移后裸库则是完全干净的空库，所以预计迁移前后使用上没有太大的差异

			9. 71 78 5 - Se aB an kN oti fic ati on Se rvi ce (in de x /p ac k 记 录 为 空 , 可 以 认 为 是 空 库)	
			10. 75 05 9 - ite m- re ba te- da sh bo ard	
			11. 75 23 4 - py bi nd _te st	
			12. 75 32 2 - SP AC EM igr ati on As sis ta nt	
			13. 75 33 4 - Py arr ow Wr ite r_p ro du ct	
			14. 75 98 7 - my sql - sy nc- cli ck ho use	

云辉建议修复代码库的原则：

- 1、如果裸库内有代码数据，则该代码库以修复为主（保证此代码库可用）
- 2、如果裸库内没有代码数据，但是，则改代码库以清理为主