Notes

Edit the date part in KOT pipeline when we move to production data.

Need to transaction timestamp in in optout config

*elifv=="TRANSACTION\_DATETIMESTAMP"*

Helper update(date)

Timestamp not added in segment and need to be removed from COT and KOT config.

Need to add transaction timestamp in in optout config

Need to update key name for 518(lytics)

479 personaKey not found

[Logs Explorer – Logging – dbce-c360-dqops-prod – Google Cloud console](https://console.cloud.google.com/logs/query;query=resource.type%3D%22dataflow_step%22%0Aresource.labels.job_id%3D%222022-07-19_19_51_36-10046840315897587583%22%0Alabels.%22compute.googleapis.com%2Fresource_name%22%3D%22registration-janrain-crm--07191951-etz9-harness-2vhw%22%0AlogName%3D%2528%22projects%2Fdbce-c360-dqops-prod-3ad4%2Flogs%2Fdataflow.googleapis.com%252Fdocker%22%20OR%20%22projects%2Fdbce-c360-dqops-prod-3ad4%2Flogs%2Fdataflow.googleapis.com%252Fharness%22%20OR%20%22projects%2Fdbce-c360-dqops-prod-3ad4%2Flogs%2Fdataflow.googleapis.com%252Fharness-startup%22%20OR%20%22projects%2Fdbce-c360-dqops-prod-3ad4%2Flogs%2Fdataflow.googleapis.com%252Fworker%22%20OR%20%22projects%2Fdbce-c360-dqops-prod-3ad4%2Flogs%2Fdataflow.googleapis.com%252Fworker-startup%22%20OR%20%22projects%2Fdbce-c360-dqops-prod-3ad4%2Flogs%2Fdataflow.googleapis.com%252Fartifact%22%2529%0A;timeRange=2022-07-20T02:51:36.898Z%2F2022-07-20T03:53:45.644Z;cursorTimestamp=2022-07-20T03:48:49.595235083Z?project=dbce-c360-dqops-prod-3ad4)- Error logs for workers not found

<https://cloud.google.com/dataflow/docs/guides/common-errors#error-syncing-pod>

['requests==2.27.1','dateparser==1.1.0','google-cloud-storage==2.0.0','google-cloud-secret-manager==2.9.2','analytics-python<=1.3.1']

"Error syncing pod, skipping" err="failed to \"StartContainer\" for \"sdk0\" with CrashLoopBackOff: \"back-off 5m0s restarting failed container=sdk0 pod=df-track-call-3b388539-07202227-ry51-harness-rd4w\_default(8ab12b647ba82bd5eb8de4ccae9248aa)\"" pod="default/df-track-call-3b388539-07202227-ry51-harness-rd4w" podUID=8ab12b647ba82bd5eb8de4ccae9248aa

*From <*[*https://teams.microsoft.com/multi-window/?agent=electron&version=22060614805*](https://teams.microsoft.com/multi-window/?agent=electron&version=22060614805)*>*

'requests==2.28.1','dateparser==1.1.1','google-cloud-storage==2.4.0','google-cloud-secret-manager==2.12.1','analytics-python<=1.3.1'

Use Git Proxy when in office:

git config --global http.proxy 172.22.218.218:8085

git config --global https.proxy 172.22.218.218:8085

Unset it when in home:

git config --global --unset http.proxy

git config --global --unset https.proxy

ghp\_WSNobykEdUBbAbB18Zr63owvjikKM9282U95

SSL Cert Error:

git config --global http.sslbackend schannel

172.22.218.218

8085

RITM4049438

IT HelpDesk- 000 800 100 7907

 Monitoring

fetch cloud\_composer\_workflow

| metric 'composer.googleapis.com/workflow/task/run\_count'

| filter

    (resource.workflow\_name == 'verify\_update\_remediation\_janrain\_optout')

| align rate(1m)

| every 1m

---------------------------------------------------------------------------

fetch global

| metric 'bigquery.googleapis.com/query/count'

| filter metric.priority = 'interactive'

fetch cloud\_composer\_workflow

| metric 'composer.googleapis.com/workflow/run\_count'

| filter

    resource.project\_id == 'dbce-c360-dqops-prod-3ad4'

    && (metric.state == 'failed')

| align rate(1m)

| every 1m

| group\_by [], [value\_run\_count\_aggregate: aggregate(value.run\_count)]

fetch bigquery\_dataset

| metric 'bigquery.googleapis.com/storage/table\_count'

| filter resource.project\_id== '1012596408563'

| align next\_older(1d)

| every 1m

| group\_by [resource.dataset\_id],

    [value\_table\_count\_aggregate: aggregate(value.table\_count)]

fetch cloud\_composer\_workflow

| metric 'composer.googleapis.com/workflow/run\_count'

| filter (metric.state== 'failed')

| align rate(1m)

| every 1m

fetch global

| metric 'bigquery.googleapis.com/slots/allocated'

| filter resource.project\_id== '790157841884'

| group\_by 1d, [value\_allocated\_max: max(value.allocated)]

| every 1m

| group\_by [metric.reservation],

    [value\_allocated\_max\_mean: mean(value\_allocated\_max)]

**KT Videos**

22-Jul-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/r/personal/m1065555_mindtree_com/Documents/Recordings/Datalake%20KT-20210722_164538-Meeting%20Recording.mp4?csf=1&web=1>

26-Jul-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/EeR7d_W-V41BlV2qRLJg0pQB8FWZozxJvNVmIf0gToFxTQ>

27-Jul-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/ERSKGbPLzWlHkV5GVfwxAmIB9Z54NOFUX5wfHN2ktRA8cg>

28-Jul-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/EU2SdGH9GWRFkC9TcMg0JQgB9oMrucJiddQOvBoSC02eng>

29-Jul-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/EfJqjfXg7ylEu0XaVMbHVyoBZ8cQfNPudk63-UYYIKaKQw>

02-Aug-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/Ecizjc3NwbJApUK3E9eB1qkBvQfExzHguzHxrK-iSx0_WQ>

03-Aug-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/Ef_cgaA9MAZKobIdDKqOoHEBqswAIQ9jGbPj_MHBGUoPzA>

04-Aug-2021:

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/EZxWfiOqs_1Pqf99PHzmBTIBj47j2lWLofd1po8RW9ABHQ>

<https://mindtreeonline-my.sharepoint.com/:v:/g/personal/m1065555_mindtree_com/ESwqKZeSIYRHskhbbXnzVSEBDe3GTCGGv9AU2OCmZEMk_Q>

05-Aug-2021:

[Recordings - OneDrive (sharepoint.com)](https://mindtreeonline-my.sharepoint.com/personal/m1065555_mindtree_com/_layouts/15/onedrive.aspx?id=%2Fpersonal%2Fm1065555%5Fmindtree%5Fcom%2FDocuments%2FRecordings%2FDatalake%20KT%2D20210805%5F170820%2DMeeting%20Recording%2Emp4&parent=%2Fpersonal%2Fm1065555%5Fmindtree%5Fcom%2FDocuments%2FRecordings&originalPath=aHR0cHM6Ly9taW5kdHJlZW9ubGluZS1teS5zaGFyZXBvaW50LmNvbS86djovZy9wZXJzb25hbC9tMTA2NTU1NV9taW5kdHJlZV9jb20vRWN2M3RlZUVfYWRLanRnRjk0ZENjb0FCYTZxbGMtTWRwdkZBbmJJcHU4MkNqUT9ydGltZT1ZQm5kTi1wYTJVZw)

**Imporatnt Links:**

* [HyperCare Validation Count](https://pgone.sharepoint.com/:x:/r/sites/External-C3PO/_layouts/15/Doc.aspx?sourcedoc=%7B874EA641-FD8A-438A-9CF4-79881C645F60%7D&file=RTDF%20Hyper%20Care%20Audience%20Counts.xlsx&wdOrigin=OFFICECOM-WEB.START.REC&ct=1627677261314&action=default&mobileredirect=true)
* [https://pgone.sharepoint.com/sites/aapgcom/Shared Documents/Forms/AllItems.aspx?id=%2Fsites%2Faapgcom%2FShared Documents%2FLegal%2F04%2E Training Materials%2FConsumer Data %2B Analytics Global Training - low resolution%2Emp4&parent=%2Fsites%2Faapgcom%2FShared Documents%2FLegal%2F04%2E Training Materials&p=true&originalPath=aHR0cHM6Ly9wZ29uZS5zaGFyZXBvaW50LmNvbS86djovcy9hYXBnY29tL0VSeXhPV19weU50SXJQZWFMZlBLU25BQmFqSnV4OHN2enQwTzRJelpfb3BhNWc\_cnRpbWU9QjZ2bW9NVmIyVWc](https://pgone.sharepoint.com/sites/aapgcom/Shared%20Documents/Forms/AllItems.aspx?id=%2Fsites%2Faapgcom%2FShared%20Documents%2FLegal%2F04%2E%20Training%20Materials%2FConsumer%20Data%20%2B%20Analytics%20Global%20Training%20%2D%20low%20resolution%2Emp4&parent=%2Fsites%2Faapgcom%2FShared%20Documents%2FLegal%2F04%2E%20Training%20Materials&p=true&originalPath=aHR0cHM6Ly9wZ29uZS5zaGFyZXBvaW50LmNvbS86djovcy9hYXBnY29tL0VSeXhPV19weU50SXJQZWFMZlBLU25BQmFqSnV4OHN2enQwTzRJelpfb3BhNWc_cnRpbWU9QjZ2bW9NVmIyVWc)

Get MPNs of required marketing programme:

select \* from `dbce-c360-segamaws-prod-9669.CDS.mpn`

**PII model report query:**

**SELECT distinct tableColumnName, \_keys, marketingProgramNumber FROM `dbce-c360-119lake-prod-36f0.r\_data\_validation.piiModelReportWeekly`**

**where \_keys NOT IN ('emailConsumerChoiceDate','emailOptStatusReason','emailDeploymentDate')**

**order by tableColumnName, \_keys, marketingProgramNumber**

**SELECT tableColumnName,key,marketingProgramNumber,description,rowCount,maxIngestdate FROM `dbce-c360-119lake-prod-36f0.r\_data\_validation.piiReportWeekly`**

**order by tableColumnName,key,marketingProgramNumber;**

Assignment group : P&G\_DBCE\_C360\_Global\_DataLakeManagers

GKE

[Mapping Kubernetes Service Accounts to GCP IAMs using Workload Identity | by Louis Vernon | Google Cloud - Community | Medium](https://medium.com/google-cloud/mapping-kubernetes-service-accounts-to-gcp-iams-using-workload-identity-b53496d543e0)

[How To Bind Kubernetes Service Account with ClusterRole - Knoldus Blogs](https://blog.knoldus.com/how-to-bind-kubernetes-service-account-with-clusterrole/)

[Cloud Wiki - Exposing Ingress Service with TLS cert (service-now.com)](https://pgglobalenterprise.service-now.com/cloudportal?id=cp_kb_article_view&sys_kb_id=185d3a1ddb6edd18ec48a07c13961950)

[Use Workload Identity  |  Google Kubernetes Engine (GKE)  |  Google Cloud](https://cloud.google.com/kubernetes-engine/docs/how-to/workload-identity#config-connector_1)

Commands:

gcloud components install kubectl

kubectl get serviceAccounts -n c360-rtdf-dev

kubectl get serviceAccounts --all-namespaces

ghp\_awx7xodBM1iWFwprlkvZOPc4kEXDDM1JjviY

ghp\_bN8dfcLMLZIXdkfbLCmJBNoImXmlls10N0XZ

git clone <https://AyushSingh1025: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@github.com/procter-gamble/rtdf_cc.git>

DNS mapping- Cloud Ops (Ip needs to be provided to cloud ops given by SRE team)

prerequite for SSO

Adding SSL cert- GKE team

1.Clone your repo to create a image

  git clone  <https://github.com/projectname/examples.git>

2.Create a new Dockerfile if it's not present in repo and mention steps for your application.

sample file for flask app

  FROM python:3.7

  RUN  mkdir /Flask\_demo\_app

  WORKDIR /Flask\_demo\_app/

  ADD . /Flask\_demo\_app/

  RUN pip install -r requirements.txt

  ENV FLASK\_APP=main.py

  ENV PORT 8080

  CMD exec gunicorn --bind :$PORT --workers 3 --threads 8 main:app

3.Run below command to create a docker image(run this command in dockerfile path)

   DOCKER\_BUILDKIT=0 docker build .

4.Once image is created you can run below command to test image locally

  docker run --rm -p 8080:8080 <image\_id>

5.Now tag your image

  docker tag <image\_id> gcr.io/<project\_id>/<image\_name>:v1

6.Push image to container registry

  docker push gcr.io/<project\_id>/<image\_name>:v1

------------------------------------------------------------------------------------------------------------------------------

|  |  |
| --- | --- |
| 1. Namespace creation by CloudOps team | RITM4343392 - DEV    RITM4390689 - PROD |
| 2. SA Creation and binding with GKE Cluster by CloudOps team | RITM4357260  - DEV  RITM4436365 - PROD |

3. KSA creation - manual

4. Annotations - manual

Docker imge push

Deploy-manual:

1.Clone your repo to create a image

  git clone  <https://github.com/projectname/examples.git>

2.Create a new Dockerfile if it's not present in repo and mention steps for your application.

sample file for flask app

  FROM python:3.7

  RUN  mkdir /Flask\_demo\_app

  WORKDIR /Flask\_demo\_app/

  ADD . /Flask\_demo\_app/

  RUN pip install -r requirements.txt

  ENV FLASK\_APP=main.py

  ENV PORT 8080

  CMD exec gunicorn --bind :$PORT --workers 3 --threads 8 main:app

3.Run below command to create a docker image(run this command in dockerfile path)

  docker build .

4.Once image is created you can run below command to test image locally

  docker run --rm -p 8080:8080 <image\_id>

5.Now tag your image

  docker tag <image\_id> gcr.io/<project\_id>/<image\_name>:v1

6.Push image to container registry

  docker push gcr.io/<project\_id>/<image\_name>:v1

Datalake cleaning

466- 2 fields data present

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 5 | 466 | questions.email | 6 | 2022-07-06 | full | 2022-07-28 | track\_complete\_tbl-properties | GBR Braun DTC |
| 6 | 466 | questions.emailSha256Hash | 6 | 2022-07-06 | full | 2022-07-28 | track\_complete\_tbl-properties | GBR Braun DTC |

-----------------------------------------

<https://pgglobalenterprise.service-now.com/dbce_homepage?id=pgcs_sc_cat_item_form&sys_id=2b261d751b135cd00e304002dd4bcb4f>- Data cleaning ticket

Requested PII Fields cleaned for MPN 19. PFA doc for reference

CREATE TABLE `dbce-c360-119lake-prod-36f0.r\_data\_validation.backup\_traits\_phoneNumber\_track\_context\_289` AS

    (

        select \* EXCEPT (dt) from

    (

        select \*,  JSON\_EXTRACT(context, '$.traits.phoneNumber') dt from `dbce-c360-119lake-prod-36f0.consumer\_raw.track\_complete\_tbl`

         where marketingProgramNumber = 289

    )

    where dt is not null and dt != "" and dt != " "

    )

CREATE TABLE `dbce-c360-119lake-prod-36f0.r\_data\_validation.formatted\_traits\_phoneNumber\_track\_context\_289` AS (

select pattern, marketingProgramNumber, userid, messageId,ingesttimeStamp,sourceId, context,

REPLACE (context , pattern, '"phoneNumber":""') replaced   from

(

    select userid, messageId, context, ingesttimeStamp,

      concat("\"phoneNumber\"",":",JSON\_EXTRACT(context, '$.traits.phoneNumber')) pattern, marketingProgramNumber, filename,

    SPLIT(filename, '/')[OFFSET(4)]sourceId, SPLIT(filename, '/')[OFFSET(5)]epoch from

    `dbce-c360-119lake-prod-36f0.consumer\_raw.track\_complete\_tbl`

    WHERE marketingProgramNumber =  289 and

    JSON\_EXTRACT(context, '$.traits.phoneNumber') is not null

    and JSON\_EXTRACT(context, '$.traits.phoneNumber') != ""

    and JSON\_EXTRACT(context, '$.traits.phoneNumber') != " "

    )

)

MERGE `dbce-c360-119lake-prod-36f0.consumer\_raw.track\_complete\_tbl` b USING

`dbce-c360-119lake-prod-36f0.r\_data\_validation.formatted\_traits\_phoneNumber\_track\_context\_289` a  ON (

    a.userid = b.userId and

 a.messageId = b.messageId and

 a.sourceId = SPLIT(b.filename, '/')[OFFSET(4)] and

a.marketingProgramNumber = b.marketingProgramNumber and

 a.ingesttimeStamp = b.ingesttimeStamp

) WHEN MATCHED THEN UPDATE SET b.context= a.replaced

*From <*[*https://teams.microsoft.com/multi-window/?agent=electron&version=22102807200*](https://teams.microsoft.com/multi-window/?agent=electron&version=22102807200)*>*