**MapReduceTask**

**GentityMapReduceTask**

* GCalMapReduceTask
  + GcalAvgHangoutMRRunner
    - GCalPercentageHangoutMapper

Lee un **GCalendarEntityData**, recupera todas las areas y por cada una de ellas genera una entrada con el código de la misma y con el **boolean hasHangout()** del **EntityData**.

* + - GcalPercentageHangoutReducer

Save entity **GcalPercentageHangoutRawData**. (Date, key (es cod. Area), total, whitHangout, avg, avg \* 100). Por cada área guarda el porcentaje de eventos del calendar que tuvieron hangout.

Emit **GcalPercentageHangoutMetric**

* + GcalendarMRRunner
    - GcalAvgMapper

Por cada área guarda un uno.

* + - GcalAvgReducer

Suma la cantidad de meetings por área, luego recupera el total de usuarios para esa misma área y realiza el avg del total de meetings / #usuarios. Se almacena en **GcalAvgRawData** (date.toUTCMillis(), key, averageOfEventsPerUser, count, numberOfUsers)

* + GcalDailyUsageMRRunner
    - GcalDailyUsageMapper

Genera por cada area un **UserDuration** el cual almacena el email del user y la duracion de la meeting.

* + - GcalDailyUsageReducer

Por cada usuario suma el tiempo total el cual estuvo en meeting. Se usa la suma siempre q esta no supere la duración de del dia laborable, sino se elige esta última. Se suman todos los totales de los usuarios para calcular el total x area y se realiza: (total del area / numero de users) / horas de dia laborable(millis) . Se almacena **GCalDailyRawData**(date.toUTCMillis(), totalTimePerArea, key, averageHours \* 100, numberOfUsers)

Emit **GCalDailyUsageMetric**

* + GcalendarPercentWithRoomMRRunner
    - GcalPercentWithRoomMapper

Por cada entitydata se almacena el codigo de area como key y si la meeting uso sala.

* + - GcalPercentWithRoomReducer

Suma el total de meetings y de las mismas que usaron sala. Calcula el % de las que usaron sala y lo almacena en **GCalPercentWithRoomRawData**(date.toUTCMillis(), key, percentageOfMeetingsWithRooms, totalCountWithRoom, totalCount)

Emit **GcalendarPercentWithRoomMetric**

* + GcalDurationRangesMRRunner
    - GcalDurationRangesMapper

Recupera todas las respuestas de aceptacion una meetings. Suma para cada area de un usuario involocrado el tipo de reunion segun su duracion, < 1h, > 2h o entre 1 y 2 hs. Genera un **GCalDurationRangesMapOutput** con el total x area para < 1h, > 2h o entre 1 y 2 hs y otro **GCalDurationRangesMapOutput** para el total de todas las areas (< 1h, > 2h o entre 1 y 2 hs).

* + - GcalDurationRangesReducer

Emit **GCalDurationRangesRawData** agrupado por key(cod area). Suma el total x area (dada todas las meetings) para < 1h, > 2h o entre 1 y 2 hs y lo almacena en **GCalDurationRangesRawData** (date.toUTCMillis(), key, lessHour, greaterTwoHours, inBetween)

* GdocMapReduceTask
  + GdocSharePercentageMRRunner
    - GdocSharePercentageMapper

Genera un entry con area(key) y **GDocShareTotal** (docsshared, docs totales) y otra entry para "total" con el mismo valor de **GdocShareTotal**.

* + - GdocSharePercentageReducer

Calcula el total de sharedDocs y totalDocs de cada area y calcula el shareDoc% (sharedDocs \* 100 / totalDocs). Almacena **GDocSharePercentageRawData**(date.toUTCMillis(), key, sharedDocs, totalDocs, sharedDocsPercentage).

Emit **GDocSharePercentageMetric**

* + GdocUserAvgNativeDocsMRRunner
    - GdocUserAvgNativeDocsMapper

Por cada entrada genera un entry con el codigo de area y un **GDocsTypeCount** con el total de documentos y el de subidos. Tmb genera un entry para total con los mismos datos.

* + - GdocUserAvgNativeDocsReducer

Genera **GDocUserAvgNativeDocsRawData**(date.toUTCMillis(), key, total, nonNative, numberOfUsers, (total - nonNative) / numberOfUsers, total / numberOfUsers)

Emit **GDocUserAvgNativeDocsMetric**

* + GdocUserAreaPercentageNativeDocsMRRunner
    - GdocUserAreaPercentageNativeDocsMapper

Calcula el numero de creados(total - uploads) en un dia(hoy - ayer). Si es mayor a 0 guarda por cada area el user q lo creo.

* + - GdocUserAreaPercentageNativeDocsReducer

Recupera el numero de usuario x area y por cada area Emit **GDocUserAreaPercentageNativeDocsRawData**(date.toUTCMillis(), key, users.size(), numberOfUsersArea, new Blob(compressedSet)). Compressedset es la lista de emails de usuarios convertida a blob.

* + GdocUserAreaPercentageContributeContentMRRunner
    - GdocUserAreaPercentageContributeContentMapper

Si la cantidad de docs nativos editados es mayor a 0 entonces por cada userArea del entity se guarda el user(mail). Cod area (key).

* + - GdocUserAreaPercentageContributeContentReducer

Emit **GdocUserAreaPercentageContributeContentRawData** (date.toUTCMillis(), key, users.size(), numberOfUsersArea, new Blob(compressedSet))

* GreportMapReduceTask
  + GmailSentReceivedCountMRRunner
    - GmailSentReceivedCountMapper

Por cada **GReportEntityData** se genera un entry von el cod de area y un **GmailSentReceivedCount**(EmailsSent, EmailsReceived). Tambien se genera un entry con total como key.

* + - GmailSentReceivedCountReducer

Se calcula el total por cada key se send an received mails. Si el # de usuarios del area es > 0 => se calcula el avg de los recibidos y los enviados y la suma de ambos. **GMailSentReceivedRawData**(date.toUTCMillis(), key, sentCount, receivedCount, numberOfUsers, averageSent, averageReceived)

Emit **GMailSentReceivedCountMetric**(key, date, averageSent, averageReceived, percentageSent, percentageReceived) donde percentage es avg / total \* 100

* GtaskMapReduceTask
  + GtaskEmailAssociatedMRRunner
    - GtaskEmailAssociatedMapper

Por cada **GTaskEntityData** se almacena un entry con cod. user y mail del usuario.

* + - GtaskEmailAssociatedReducer

Se calcula el porcentaje de # de usuarios de un area que son dueños de tareas con respecto al el # total de user del area. **GTaskEmailAssociatedRawData**(date.toUTCMillis(), key, numberOfUsers, users.size(), percentageOfUsers)

Emit **GtaskEmailAssociatedPercentageMetric**

* UserAreaCountTask
  + UserAreasCountMRRunner
    - UserAreasCountMapper

Cuenta los user por area.

* + - UserAreasCountReducer

Emit Cuenta los user por area.

**MetricStatsMapReduceTask**

* GdocSharePercentageMetricStatMRRunner
  + GdocSharePercentageMetricStatMapper
  + GdocSharePercentageMetricStatReducer

Calcula la media y la desviacion de los gdoc compartidos por el total de los usuarios

* *GcalDailyUsageMetricStatMRRunner*
  + *GcalDailyUsageMetricStatReducer*

Media y desviación del tiempo en uso en calendar por el total de los usuarios.

* GcalendarAvgMetricStatMRRunner

Media y desviación de los eventos del calendar por el total de los usuarios.

* GdocUserAvgNativeDocsMetricStatMRRunner

Calcula la med y devstan de la diferencia entre total y nativos para el total de los usuarios.

* GcalendarPercentWithRoomMetricStatMRRunner

Calcula la media del # de citas con sala / el # de citas totales y la StdDev de lo mismo.

* GcalPercentageHangoutMetricStatMRRunner
* GTaskEmailAssociatedPercentageMetricStatMRRunner
* GmailSentReceivedCountMetricStatMRRunner
* GdocUserAreaPercentageNativeDocsMetricStatMRRunner
* GdocUserAreaPercentageContributeContentMetricStatMRRunner

**PopulationMapReduceTask** (genera datos para los usuarios en particular y no con areas)

* GcalendarAvgPopulationMRRunner
* GcalendarPercentWithRoomPopulationMRRunner
* GcalDailyUsagePopulationMRRunner
* GcalAvgHangoutPopulationMRRunner
* GmailSentReceivedCountPopulationMRRunner
* GtaskEmailAssociatedPopulationMRRunner
* GdocSharePercentagePopulationMRRunner
* GdocUserAvgNativeDocsPopulationMRRunner
* GdocUserAreaPercentageNativeDocsPopulationMRRunner
* GdocUserAreaPercentageContributeContentPopulationMRRunner

**RepeatedAreaDescMRTask**

* RepeatedAreaDescMRRunner

**UserAreasTask**

* UserAreasMRRunner