Federated GMQL queries

Introduction:

Here, we present multiple versions of the example query described below, to show possible execution strategies of a Federated GMQL query. All query versions are ready to be used on the dedicated server (GeCo as LOCAL).

In this example query, we show a case study on Adenoid Cystic Carcinoma (ACC), an uncommon form of malignant neoplasm that arises within secretory glands of the head and neck. The researcher has a dataset of several MUTATION samples of ACC patients in her private GMQL instance, and she likes to know which are in ACC the highly expressed and highly mutated genes that are associated with MYC transcription factor.

She integrates her experimental dataset with public data from TCGA and ENCODE, available in the CINECA and DEIB GMQL instances respectively, by performing a Federated GMQL query on these data. This case study shows the relevance of the Federated GMQL system and also the expressive power of GMQL in building queries of biological interest.

We present 4 *distributed* execution strategies (DIST-1 to DIST-4), 3 *centralized* ones (CENT-1 to CENT-3), and the BEST strategy, according to the running performances on the mentioned GMQL instances available. We also show 1 example for the *protective* directive and 2 examples for the *policy* directive (1 for *distributed* and 1 for *centralized* policy, respectively),

Distributed 1:

In this example, all the unary GMQL operations are on the GMQL instance/machine where the dataset is selected. The binary GMQL operations, JOIN and MAP, are both executed on the DEIB instance.

```
######################## DIST-1 (JOIN, MAP: DEIB) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                 CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:CINECA)
AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:CINECA) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                      DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:DEIB) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:LOCAL) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:DEIB) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:DEIB) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## DIST-1 (JOIN, MAP: DEIB) ########################
```

Distributed 2:

In this example, all the unary GMQL operations are on the GMQL instance/machine where the dataset is selected. The binary GMQL operations, JOIN and MAP, are both executed on the CINECA instance.

```
######################## DIST-2 (JOIN, MAP: CINECA) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                 CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:CINECA)
AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:CINECA) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:CINECA) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:LOCAL) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:CINECA) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:CINECA) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## DIST-2 (JOIN, MAP: CINECA) ########################
```

Distributed 3:

In this example, all the GMQL unary operations are on the GMQL instance/machine where the dataset is selected. The binary GMQL operations, JOIN and MAP, are executed on the DEIB and GeCo (LOCAL) instance, respectively.

```
######################## DIST-3 (JOIN: DEIB, MAP: GeCo) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                 CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:CINECA)
AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:CINECA) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:DEIB) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:LOCAL) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:LOCAL) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:LOCAL) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## DIST-3 (JOIN: DEIB, MAP: GeCo) ########################
```

Distributed 4:

In this example, all the unary GMQL operations are on the GMQL instance/machine where the dataset is selected. The binary GMQL operations, JOIN and MAP, are executed on the CINECA and GeCo (LOCAL) instances, respectively.

```
######################## DIST-4 (JOIN: CINECA, MAP: GeCo) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                 CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:CINECA)
AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:CINECA) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:CINECA) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:LOCAL) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:LOCAL) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:LOCAL) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## DIST-4 (JOIN: CINECA, MAP: GeCo) ########################
```

Centralized 1:

In this example, all the GMQL selection operations run on the GMQL instance/machine where the dataset is selected. All the other operations run on the DEIB instance.

```
######################## CENT-1 (DEIB) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:DEIB) AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:DEIB) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:DEIB) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:DEIB) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:DEIB) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:DEIB) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## CENT-1 (DEIB) ########################
```

Centralized 2:

In this example, all the GMQL selection operations run on the GMQL instance/machine where the dataset is selected. All the other operations run on the CINECA instance.

```
######################## CENT-2 (CINECA) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:CINECA)
AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:CINECA) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                      DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:CINECA) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:CINECA) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:CINECA) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:CINECA) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## CENT-2 (CINECA) ########################
```

Centralized 3:

In this example, all the GMQL selection operations run on the GMQL instance/machine where the dataset is selected. All the other operations run on the GeCo (LOCAL) instance.

```
######################## CENT-3 (GeCo) ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:LOCAL) AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:LOCAL) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:LOCAL) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:LOCAL) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:LOCAL) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:LOCAL) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## CENT-3 (GeCo) ########################
```

Best query:

In this example, all the GMQL selection operations run on the GMQL instance/machine where the dataset is selected, and the COVER operation run on the CINECA instance, i.e., where the covered dataset is selected. All the other operations run on the GeCo (LOCAL) instance.

```
######################## BEST ########################

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                 CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count); at:CINECA)
AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000; at:LOCAL) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct; at:LOCAL) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE(at:LOCAL) myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count; at:LOCAL) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0; at:LOCAL) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## BEST ########################
```

Protected dataset:

In this example, we set as *protected* the local dataset `Example_Mutation` and leave without any specific setting the execution location of all GMQL operations, but the selection of the input datasets, which the system choses according to the default execution policy, i.e., the *distributed* policy. This guaranties that the *protected* dataset will not be moved to other GMQL instances, keeping it private on the local GMQL instance/machine.

```
######################## Protected ########################
@protected Example_Mutation

#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count)) AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE() myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## Protected ########################
```

Distributed policy:

In this example, we set only the location of the execution of the GMQL operations of selection of the input datasets, and specify the *distributed* policy, which is the default one, for the execution of all the other operations.

```
######################## Distributed policy ########################
@policy distributed
#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                 CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count)) AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE() myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## Distributed policy ########################
```

Centralized policy:

In this example, we set only the location of the execution of the GMQL operations of selection of the input datasets, and specify the *centralized* policy at the DEIB GMQL instance for the execution of all the other operations. Thus, the query is equivalent to the Centralized 1. By changing the centralized location into CINECA or LOCAL, we can have a query execution equivalent to the Centralized 2 or Centralized 3, respectively.

```
######################## Centralized policy at DEIB ########################
@policy centralized DEIB
#  1
AccRnaseq = SELECT(manually_curated__tumor_tag == "acc" AND
                   manually_curated__tissue_status == "tumoral"; at:CINECA)
                                                CINECA.HG19_TCGA_rnaseqv2_gene;

#  2
AccExp = COVER(1, ANY; aggregate: mean_exp as AVG(normalized_count)) AccRnaseq;

#  3
AccExpFilt = SELECT(region: mean_exp > 1000) AccExp;

#  4
Myc = SELECT(gcm_curated__cell_line == "H1-hESC" AND
             target__name == "MYC-human" AND
             file__output_type == "conservative idr thresholded peaks"; at:DEIB)
                                                    DEIB.HG19_ENCODE_NARROW_2019_01;

#  5
GeneMyc = JOIN(dist < 0; output: left_distinct) AccExpFilt Myc;

#  6
myMutation = SELECT(at:LOCAL) Example_Mutation;

#  7
myMutationMerge = MERGE() myMutation;

#  8
GeneMycMut= MAP(count_name: mut_count) GeneMyc myMutationMerge;

#  9
ResGenes = SELECT(region:mut_count > 0) GeneMycMut;

# 10
MATERIALIZE ResGenes INTO ResGenes;

######################## Centralized policy at DEIB ########################
```