POLITECNICO
MILANO 1863

Distributed Processing and Optimisations for Genomic Data

Andrea Gulino

PHD CANDIDATE - XXXIII CYCLE (3rd YEAR)

Thesis-related Publications

1 🟦 A Kaitoua, A Gulino, M Masseroli, P Pinoli, S Ceri
Scalable genomic data management system on the cloud
2017 International Conference on High Performance Computing & Simulation (HPCS).

2 🟦 S. Ceri, A. Bernasconi, A. Canakoglu, A. Gulino, A. Kaitoua. M. Masseroli, L. Nanni, P. Pinoli
A project for exploring and integrating signals from the genome
2017 International Conference on Data Analytics and Management in Data Intensive Domains.

3 🟦 **A. Gulino**, A. Kaitoua, S. Ceri
Optimal Binning for Genomics
IEEE Transactions on Computers 68 (1), 125-138

4 🟦 M. Masseroli et al.
Processing of big heterogeneous genomic datasets for tertiary analysis of NGS data
2018 - Bioinformatics

5 🟦 S. Ceri et al.
Demonstration of GenoMetric Query Language
2018 - Proceedings of the 27th ACM International Conference on Information and Knowledge

6 🟨 A. Canakoglu, P. Pinoli , A. Gulino, L. Nanni, M. Masseroli, S. Ceri
Federated sharing and processing of genomic datasets for tertiary data analysis
Bioinformatics (Journal)

7 🟪 A. Gulino, A. Canakoglu, S. Ceri
Performance Prediction of Scientific Workflows on Spark.
??

🟦 **PUBLISHED**
🟨 **SUBMITTED**
🟪 **WRITING**
🟩 1st Author

Other Publications

1 🟦 (E. Stamoulakatou, **A. Gulino**), P. Pinoli
DLA: a Distributed … Algorithm for Biological Sequence Pattern Mining
2018 IEEE International Conference on Big Data (Big Data), 1121-1126 Management

2 🟦 **A. Gulino**, E. Stamoulakatou, A. Canakoglu, P. Pinoli
Analysis and Visualization of Mutation Enrichments for Selected Genomic Regions and Cancer Types
IEEE 2019 BIBM (Conference)

3 🟨 A. Gulino, S. Ceri, G. Gottlob, E. Sallinger, L. Bellomarini
Distributed Company Control Problem
SIGMOD 2020

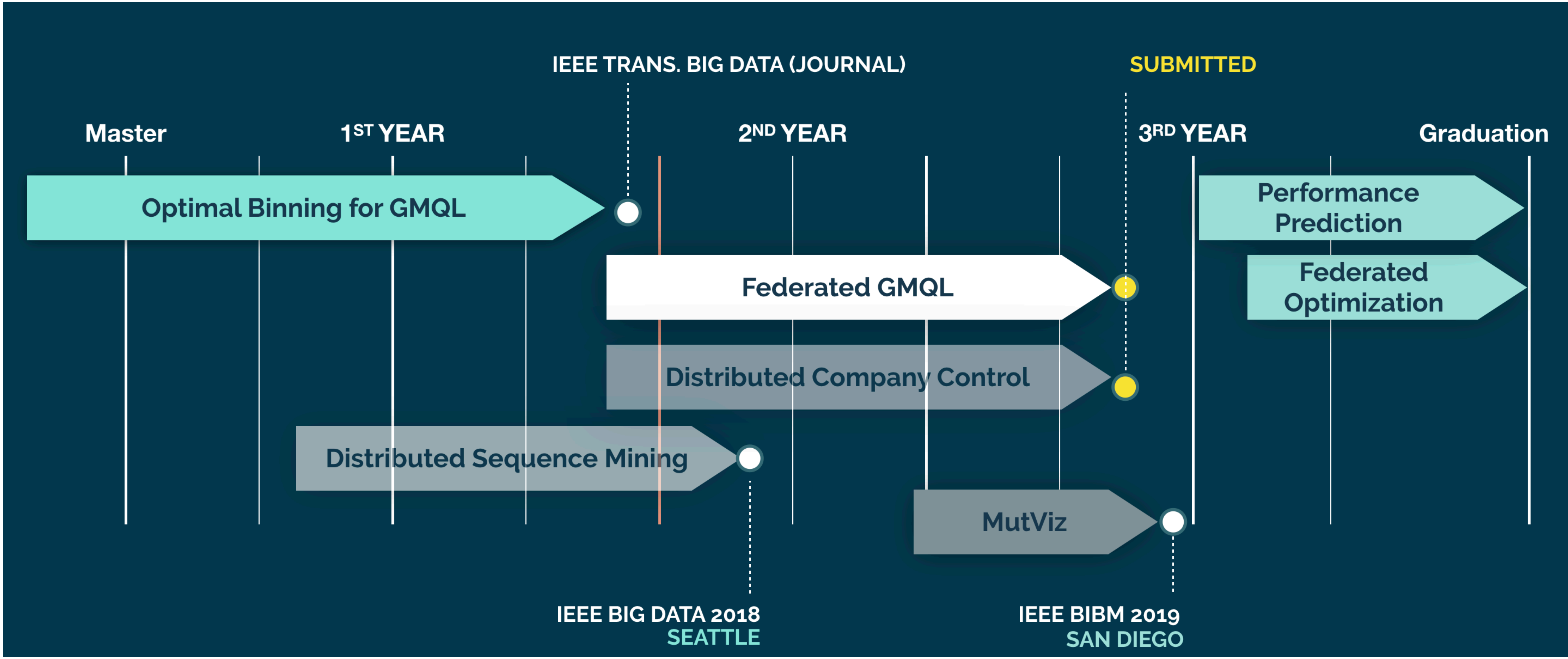
🟦 **PUBLISHED**

🟨 **SUBMITTED**

🟪 **WRITING**

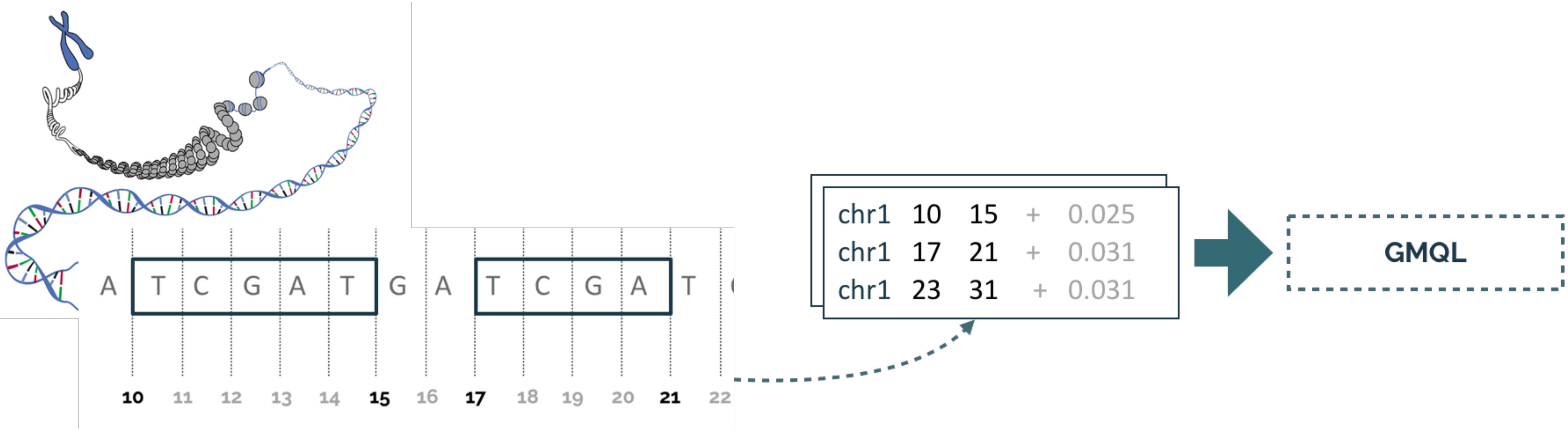
🟩 1st Author

Research Timeline



Genomic Data

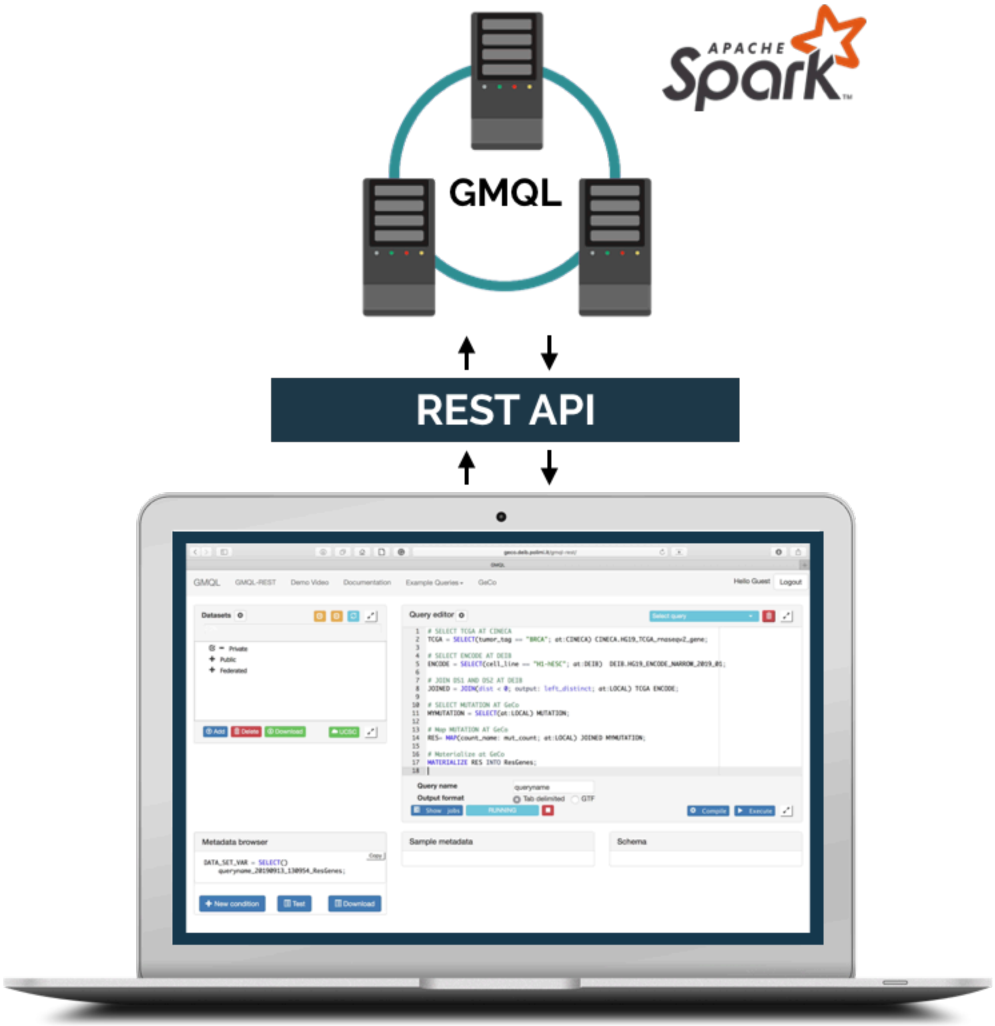
- Genome sequencing is a prolific **source of big data**.
- Opens new opportunities for biological research.
- DNA regions can be represented in a file by means of their **start** and **stop** positions.
- **Genomic Datasets** are **big** collections of such files.



```
chr1  10  15  +  0.025
chr1  17  21  +  0.031
chr1  23  31  +  0.031
```

GMQL (GenoMetric Query Language)

- A language for **querying big genomic datasets.**
- Standard operations (SQL-like) + **domain-specific operations** for **mapping** and **joining** billions of genomic regions.
- Cloud-computing processing system based on **Apache Spark**.
- Publicly accessible through a user-friendly web interface.

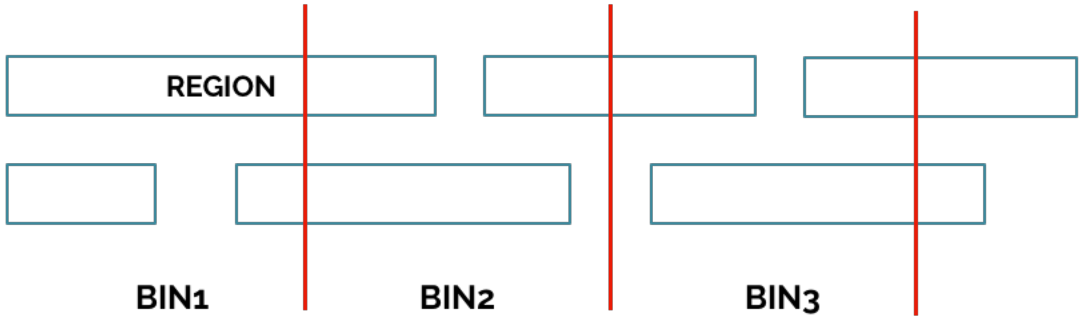


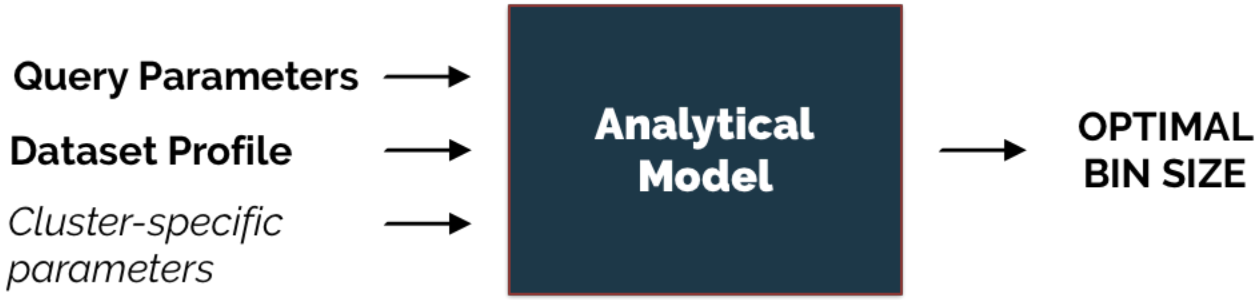
```
# SELECT TCGA AT CINECA
TCGA = SELECT(tumor_tag == "BRCA"; at:CINECA) CINECA.HG19_TCGA_rnaseqv2_gene;

# SELECT ENCODE AT DEIB
ENCODE = SELECT(cell_line == "H1-hESC"; at:DEIB) DEIB.HG19_ENCODE_NARROW_2019_01;

# JOIN DS1 AND DS2 AT DEIB
JOINED = JOIN(dist < 0; output: left_distinct; at:LOCAL) TCGA ENCODE;

# SELECT MUTATION AT GeCo
MYMUTATION = SELECT(at:LOCAL) MUTATION;

# Map MUTATION AT GeCo
RES= MAP(count_name: mut_count; at:LOCAL) JOINED MYMUTATION;

# Materialize at GeCo
MATERIALIZE RES INTO ResGenes;
```

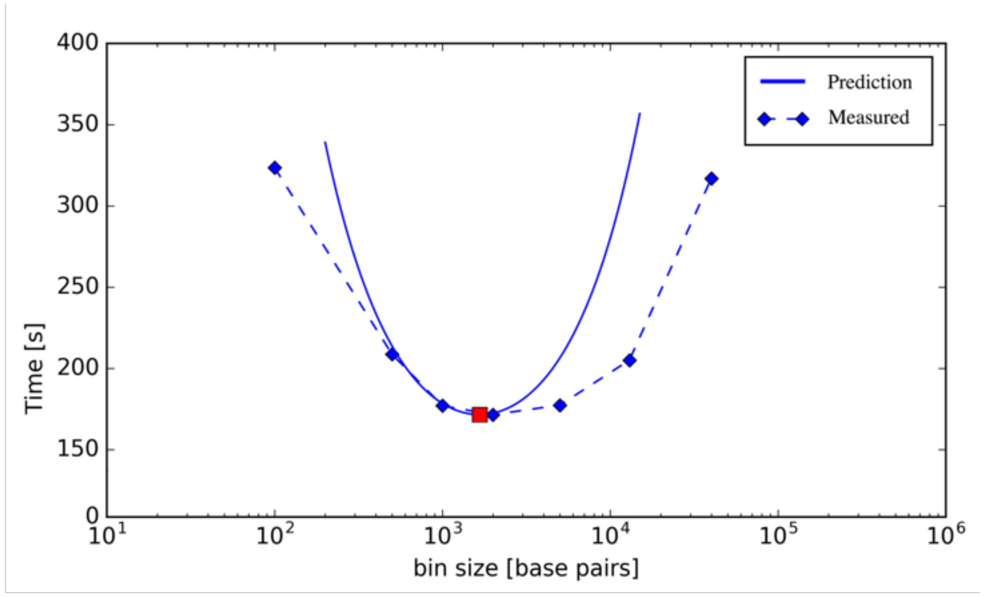
```
DATA_SET_VAR = SELECT()
    queryname_20190913_130954_ResGenes;
```

Optimal Binning (1st year)

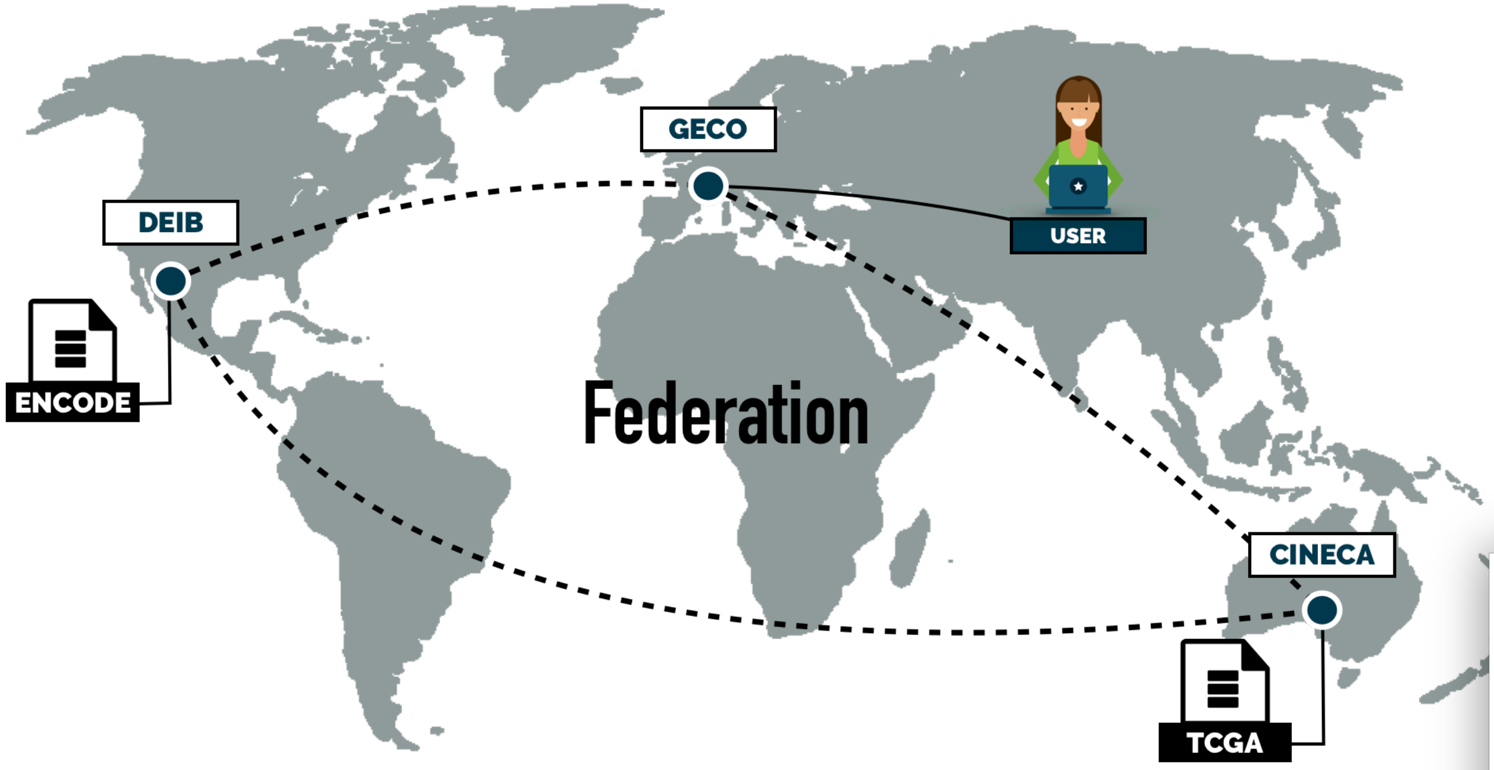
Binning splits the genomic space into portions (bins) of equal size so as to enable parallel processing of regions in different bins.

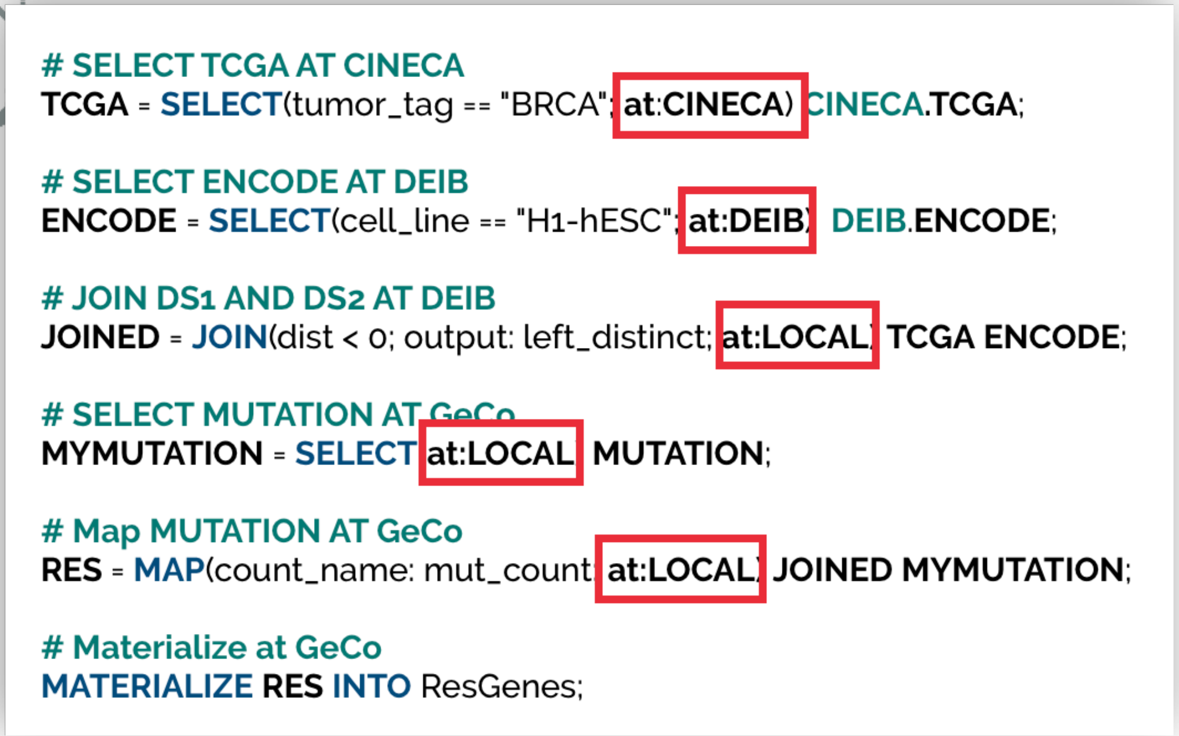






Federated GMQL (2nd year)





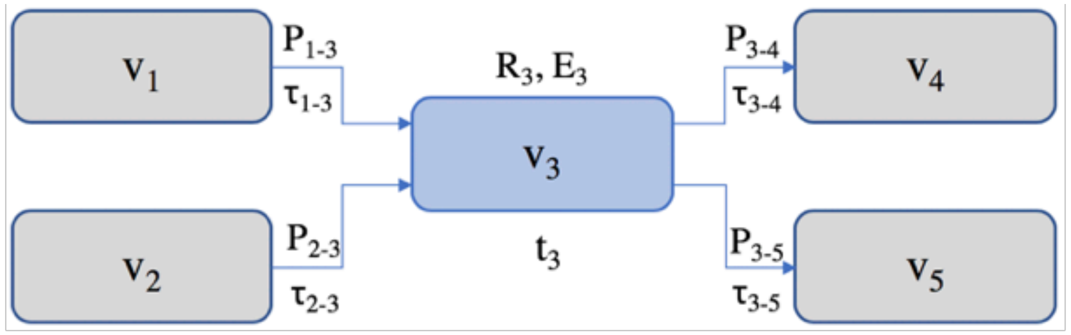
```
# SELECT TCGA AT CINECA
TCGA = SELECT(tumor_tag == "BRCA"; at:CINECA) CINECA.TCGA;

# SELECT ENCODE AT DEIB
ENCODE = SELECT(cell_line == "H1-hESC"; at:DEIB)  DEIB.ENCODE;

# JOIN DS1 AND DS2 AT DEIB
JOINED = JOIN(dist < 0; output: left_distinct; at:LOCAL) TCGA ENCODE;

# SELECT MUTATION AT GeCo
MYMUTATION = SELECT(at:LOCAL) MUTATION;

# Map MUTATION AT GeCo
RES = MAP(count_name: mut_count; at:LOCAL) JOINED MYMUTATION;

# Materialize at GeCo
MATERIALIZE RES INTO ResGenes;
```

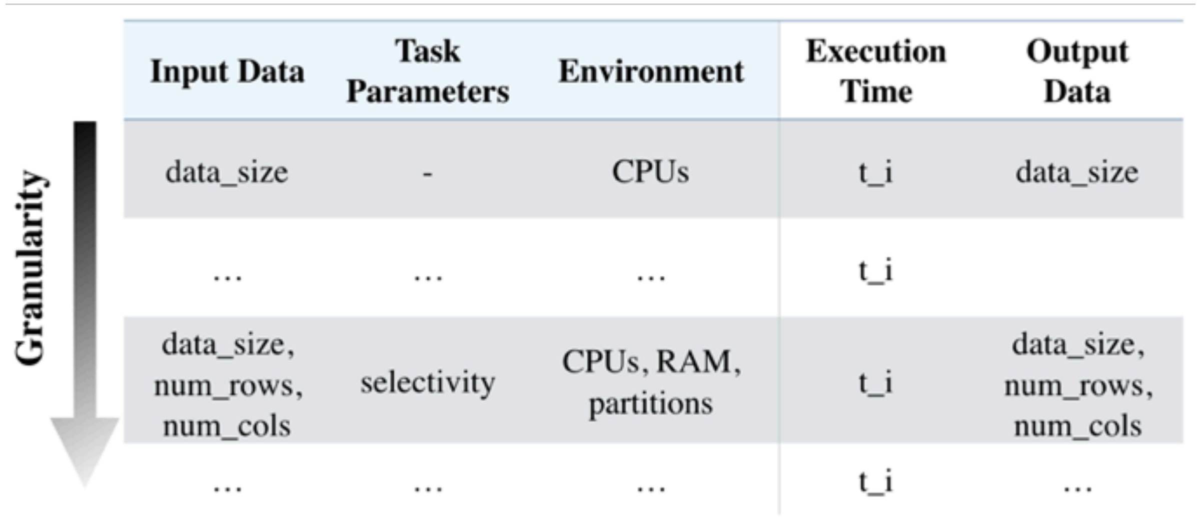
Performance Prediction (ongoing)

- Objective: use ML to model the execution time of each GMQL operator and estimate the response time of an entire GMQL query (DAG).

- In general, we would like to predict the execution time of Scientific Workflows implemented on cloud frameworks (e.g. Spark).

- Exploiting those models:
    - we can predict optimal execution parameters (e.g. bin size, number of CPUs, memory)
    - we can produce optimal query plans for Federated GMQL queries.




Granularity

Input Data	Task Parameters	Environment	Execution Time	Output Data
data_size	-	CPUs	t_i	data_size
…	…	…	t_i	
data_size, num_rows, num_cols	selectivity	CPUs, RAM, partitions	t_i	data_size, num_rows, num_cols
…	…	…	t_i	…