VANTAGE

From theory to practice

Running a PET analysis (without programming) on vantage6

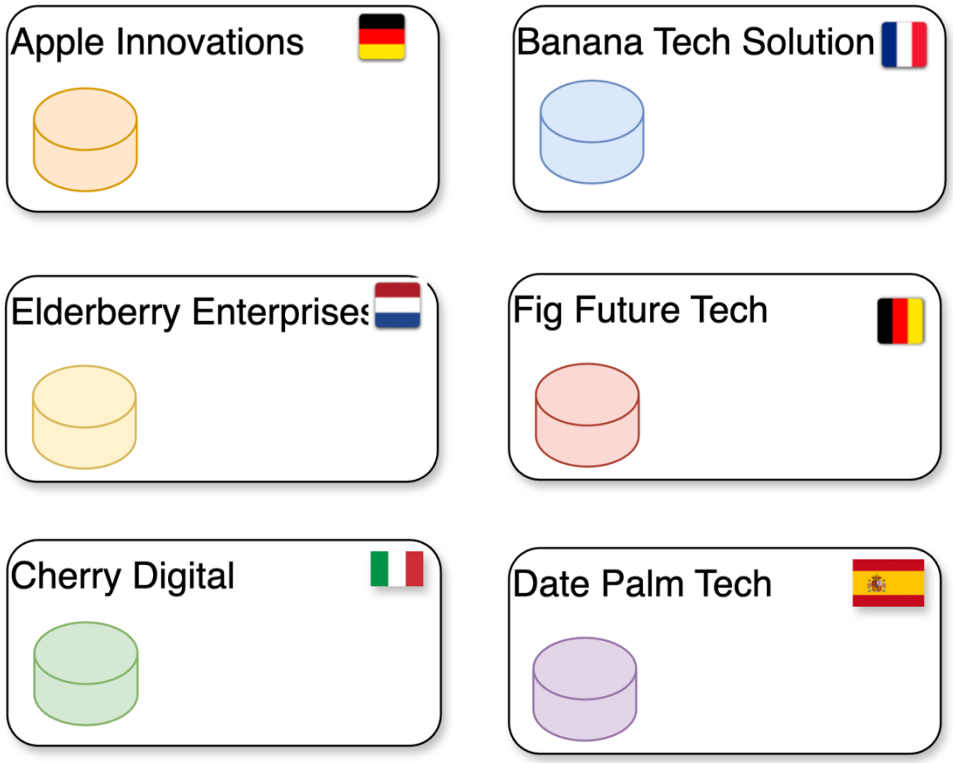
netherlands eScience center

KNL integraal kankercentrum Nederland

A theoretical scenario

VANTAGE

- An international consortium of research organizations working together on aging research.

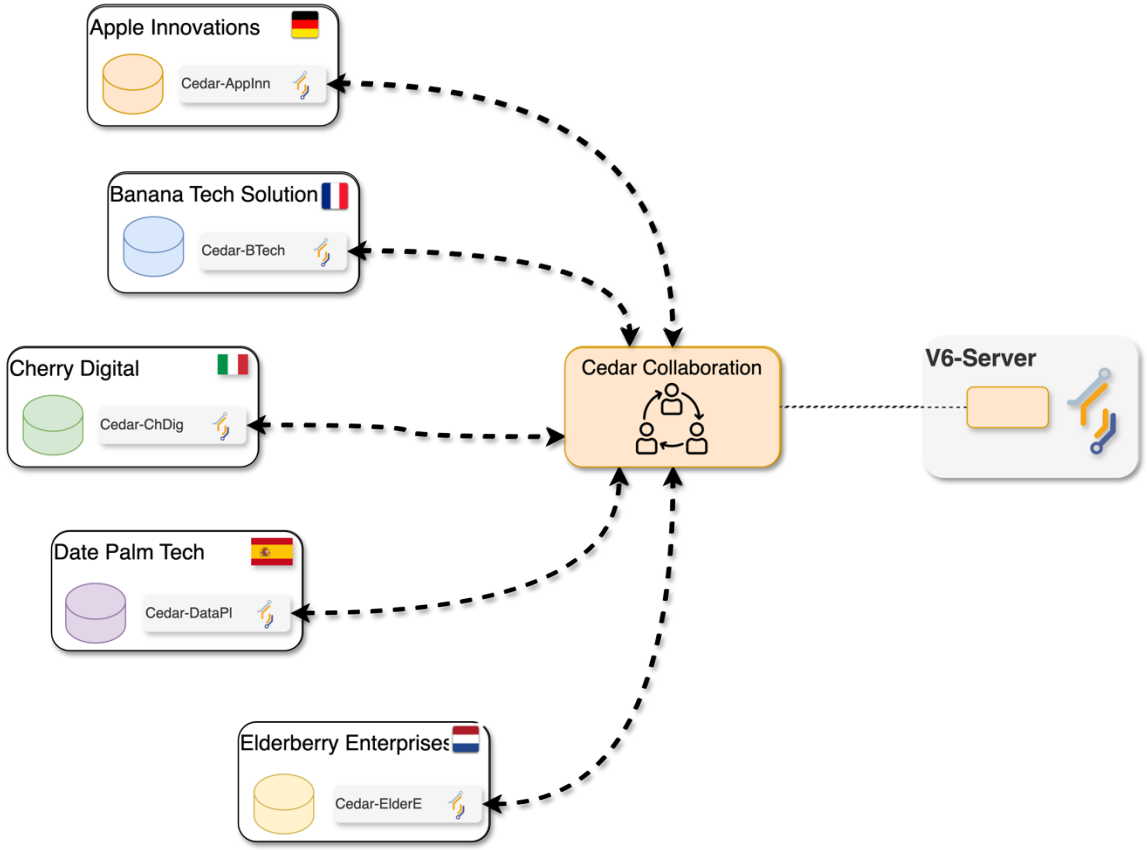


gender	age	height	weight	isOverweight	ageGroup
M	39	152	108	False	30 - 40
M	8	118	106	False	0 - 10
M	16	161	110	True	10 - 20
M	94	110	115	True	90 - 100
M	47	117	152	True	40 - 50
F	29	127	110	True	20 - 30
M	5	95	65	False	0 - 10
M	39	142	196	False	30 - 40
F	20	189	112	False	20 - 30
F	84	145	116	False	80 - 90

netherlands eScience center

IKNL integraal kankercentrum Nederland

The scenario – checking v6 concepts!



- Six organizations, one collaboration.
- How many nodes need to be set up?

Node

- Where are these nodes installed?

The scenario – checking v6 concepts!

VANTAGE

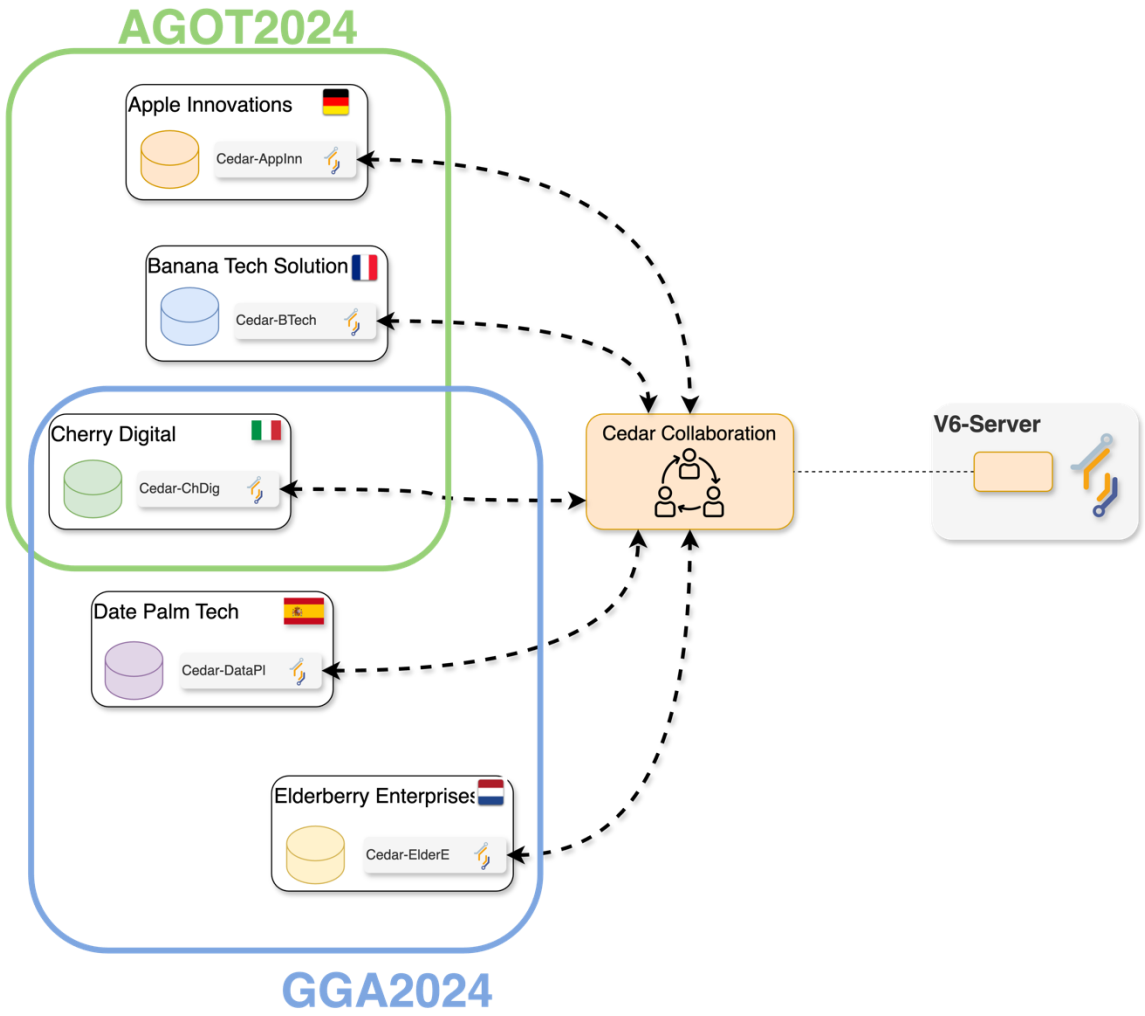
The consortium (also known as the collaboration) will initially focus on two different projects, each one involving different organizations.

- *Age-Related Variations in Overweight Prevalence: A Comparative Analysis Across Gender and Age Groups* **(AGOT2024)** : *four organization with relevant data.*

- *The Effect of Gender on Height Development Across Various Age Groups* **(GGA2024)**: *three organization with relevant data.*

Which V6 concept is applicable here?

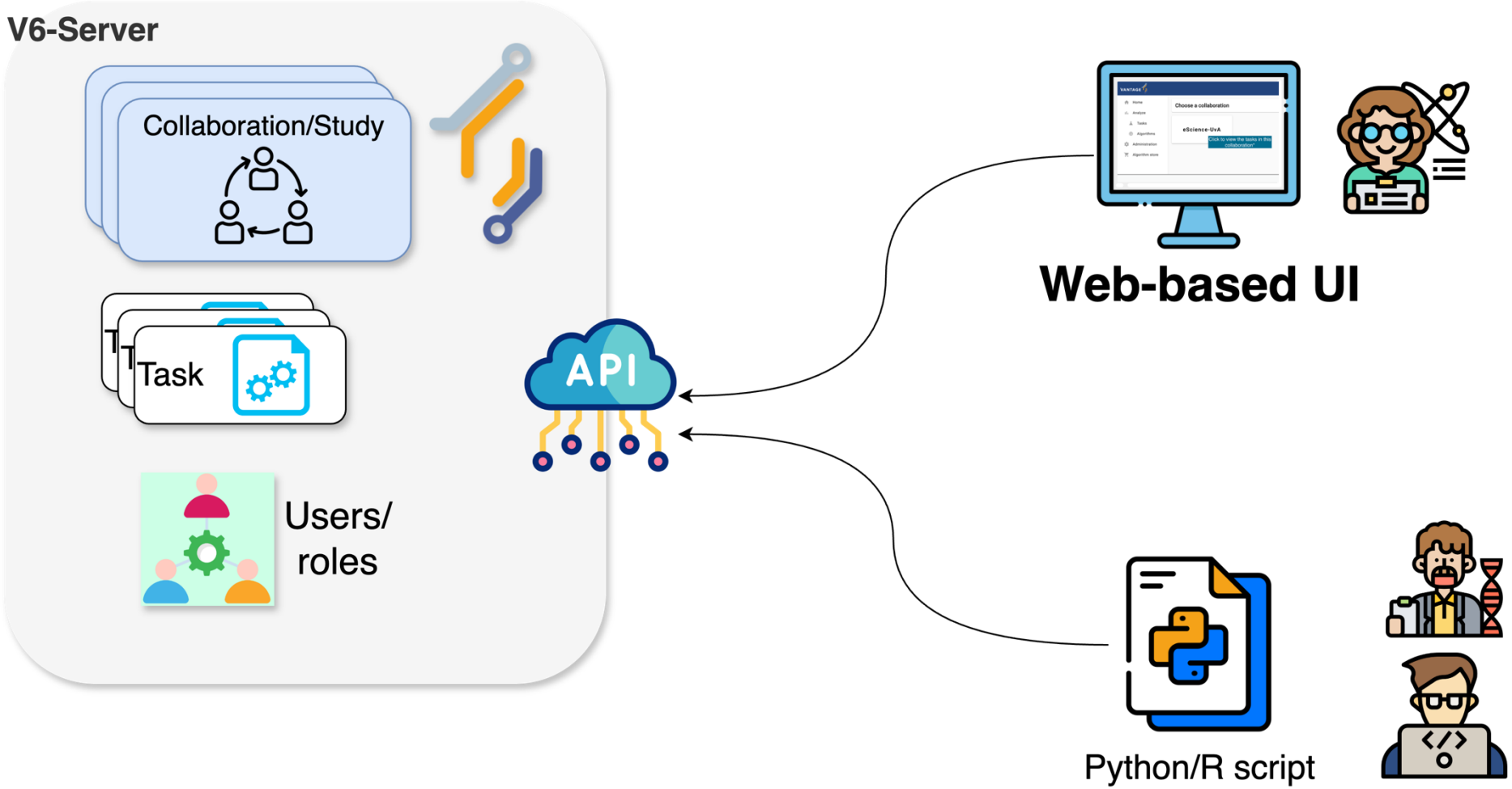
The scenario – checking v6 concepts!



- Two studies: GGA2024, AGOT2024.

How to conduct an analysis?

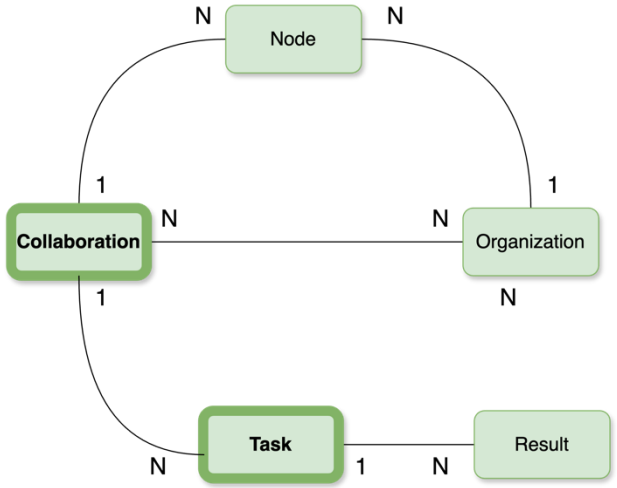
VANTAGE

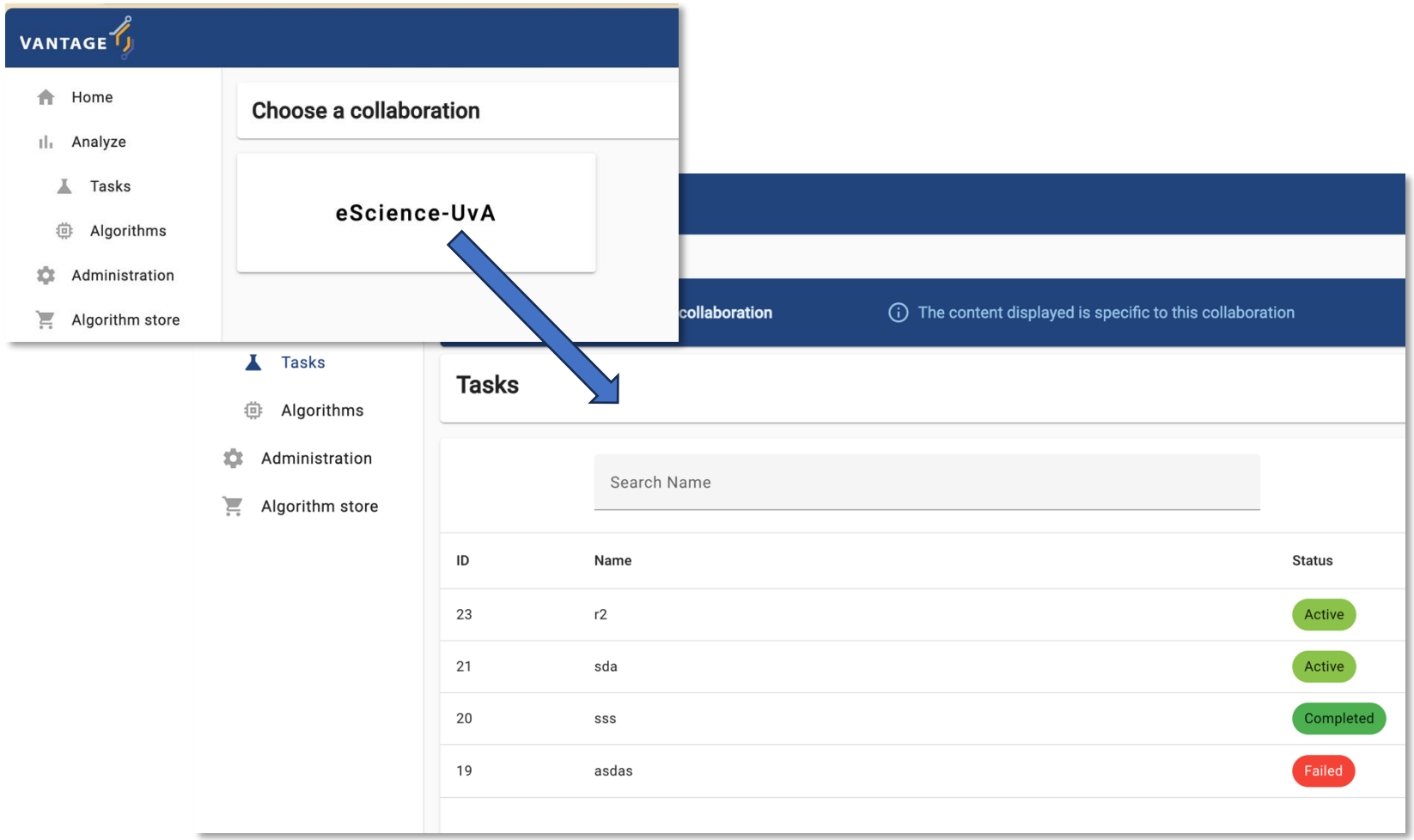


netherlands eScience center

IKNL integraal kankercentrum Nederland

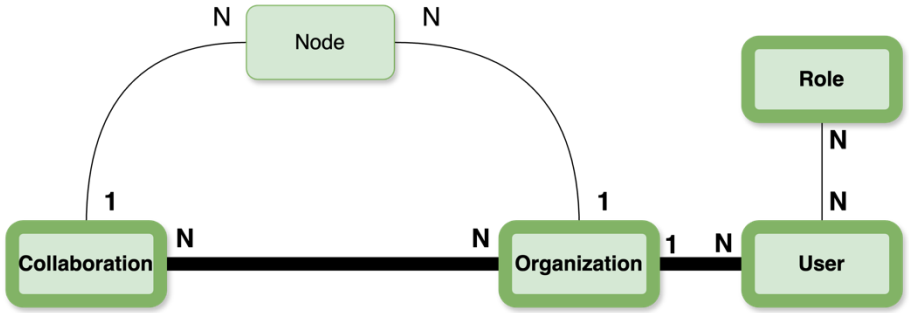
In this episode: web-based UI

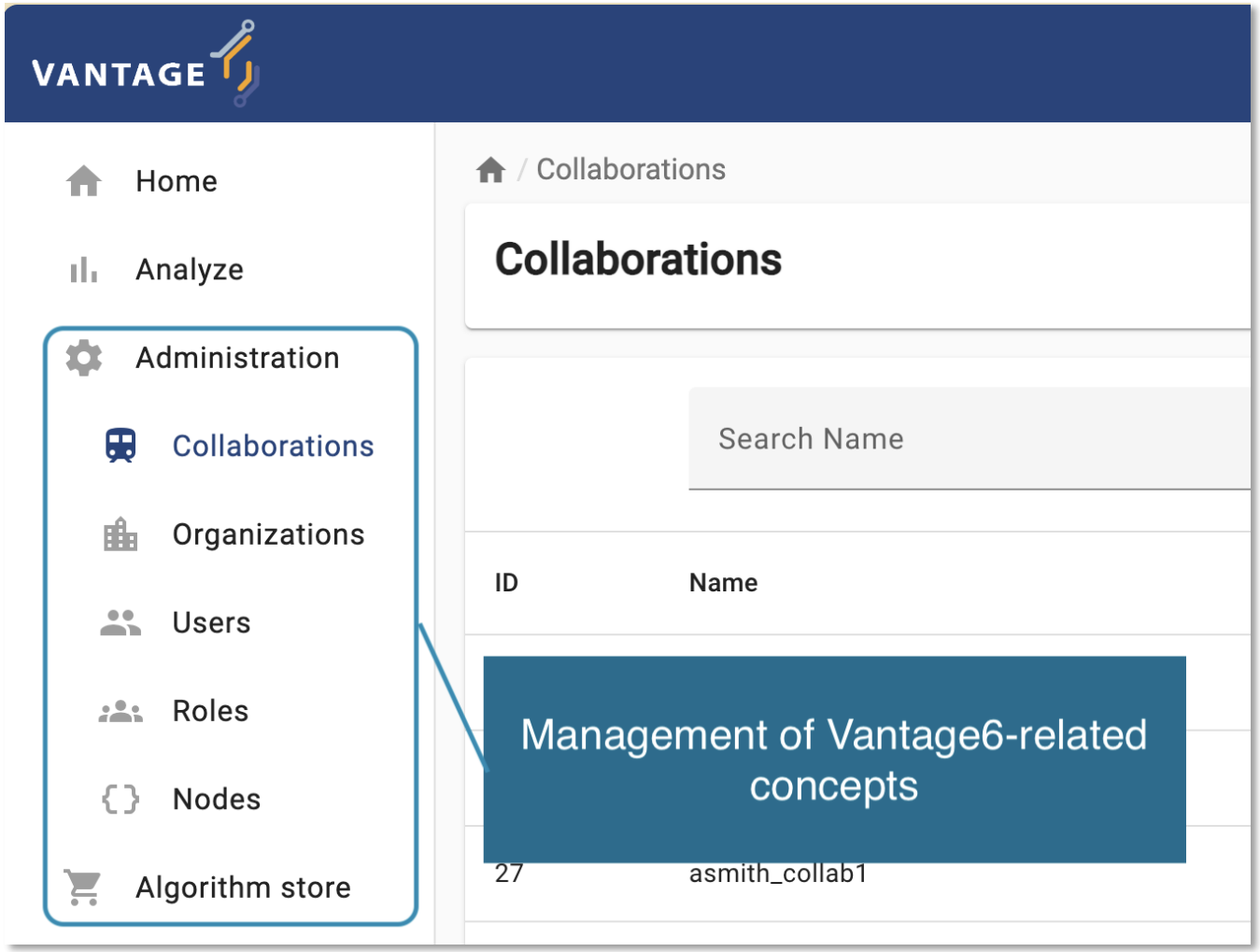




Web-based UI

VANTAGE





netherlands eScience center

KNL integraal kankercentrum Nederland

Getting familiar with the UI

VANTAGE

CHALLENGE

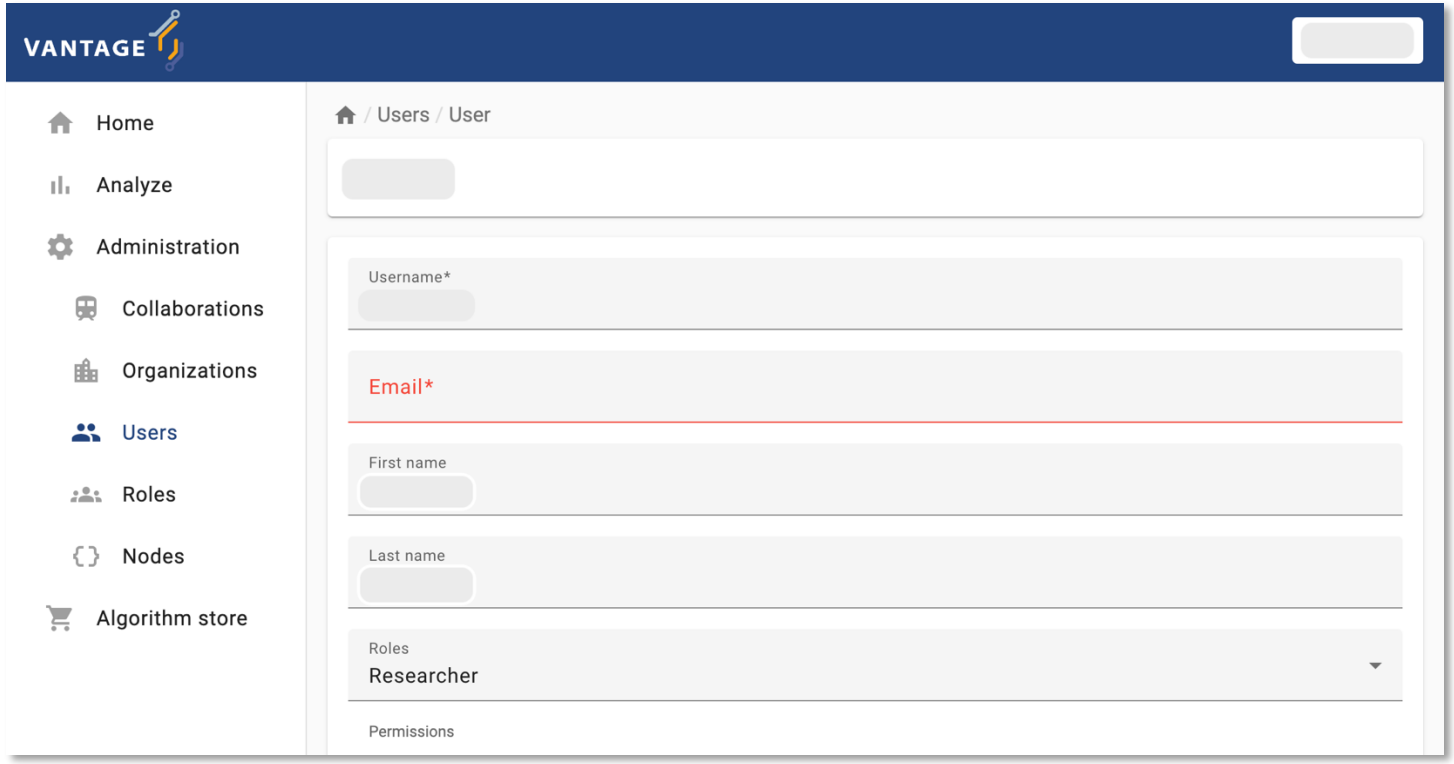
Log into the UI using the information provided and navigate to the administration page. Then, update your email, first name, and last name.

HINT
Use the **Users** option on the Administration panel on the left.

Getting familiar with the UI

VANTAGE

SOLUTION



netherlands eScience center

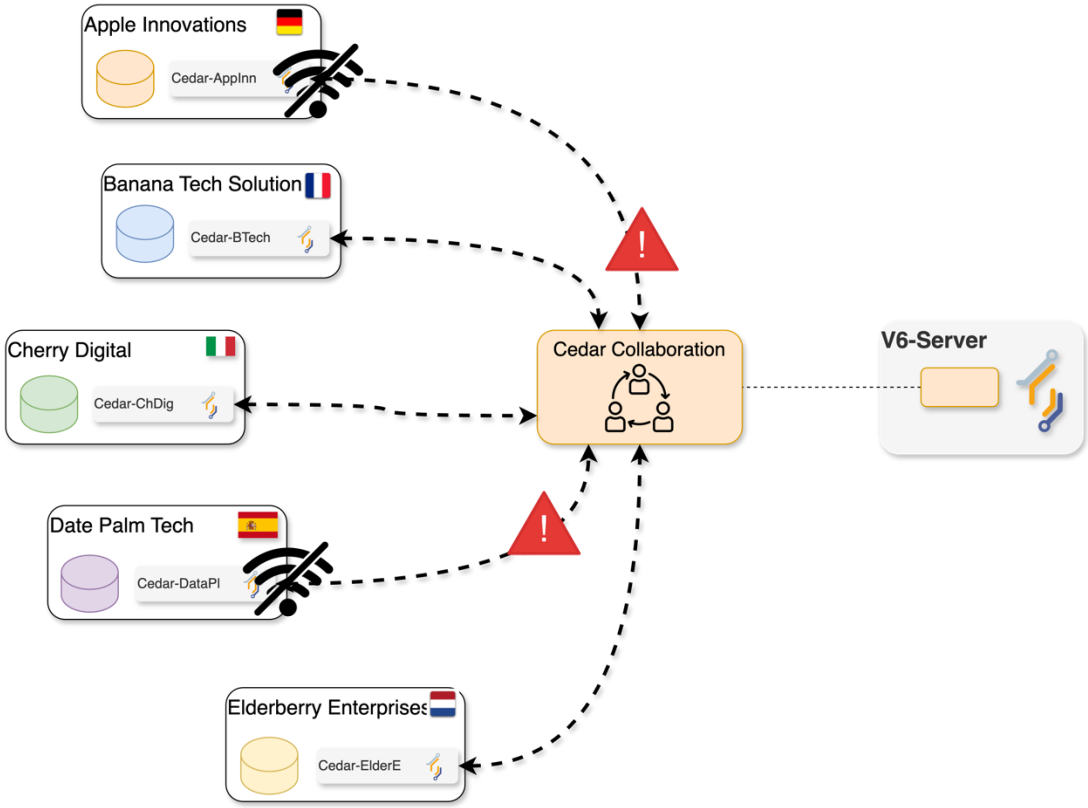
KNL integraal kankercentrum Nederland

Running a PET: federated analysis

VANTAGE

Are the nodes on my collaboration or study ready for a federated analysis?

- Node within a collaboration: autonomously managed by its corresponding organization.
- Some could be offline.
- UI is handy for checking this!



Running a PET: federated analysis

VANTAGE

CHALLENGE

With your researcher credentials, explore the collaboration you have access to. Check which organizations are part of it and if they are online. Then, check which organizations were assigned to each study: **AGOT2024**, **GGA2924**.

1. Which study is ready for a federated analysis?
2. If you need to perform an analysis for the study that is **not** ready, which organization you would need to contact to fix this situation?

HINT

Use the **Collaborations** option on the Administration panel on the left. A hand-drawn diagram of your collaboration and studies will be handy for the next steps!

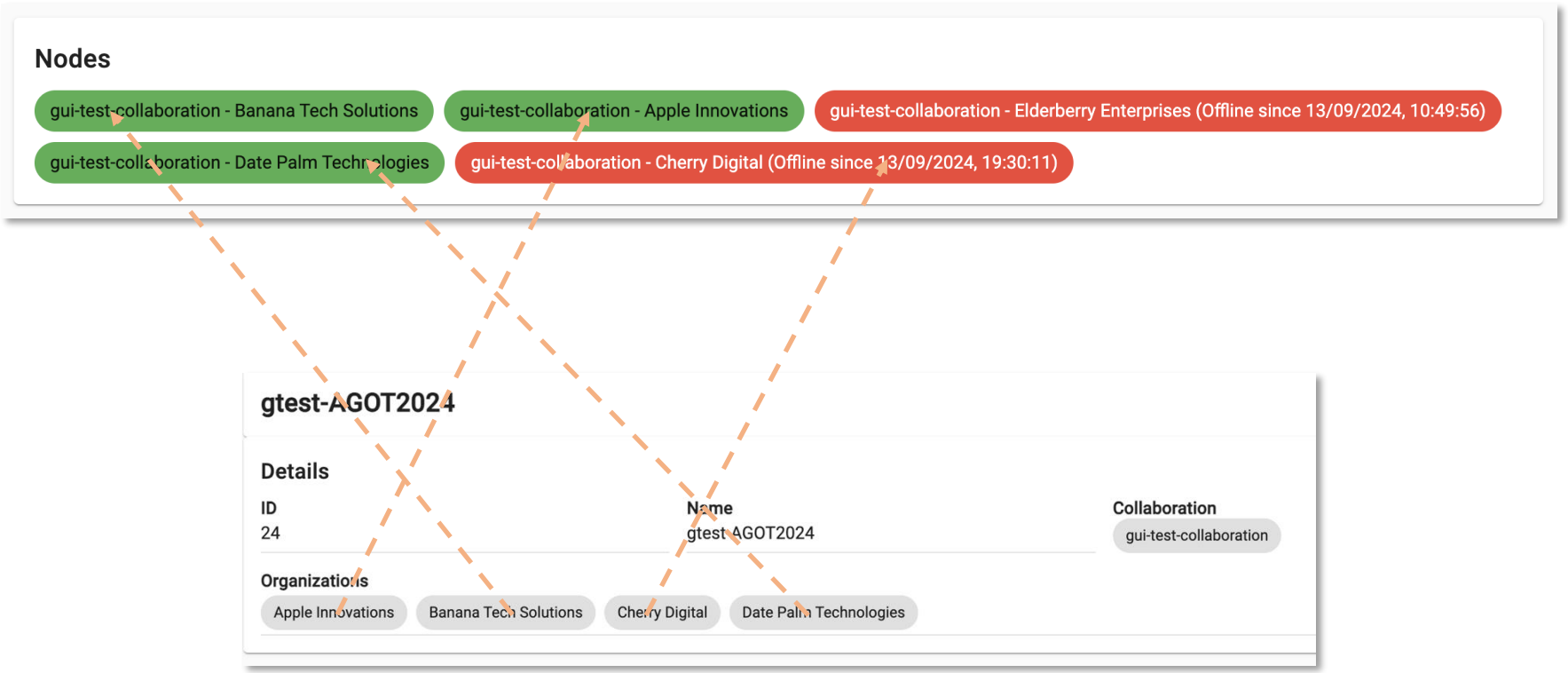
netherlands eScience center

IKNL integraal kankercentrum Nederland

Getting familiar with the UI

VANTAGE

SOLUTION

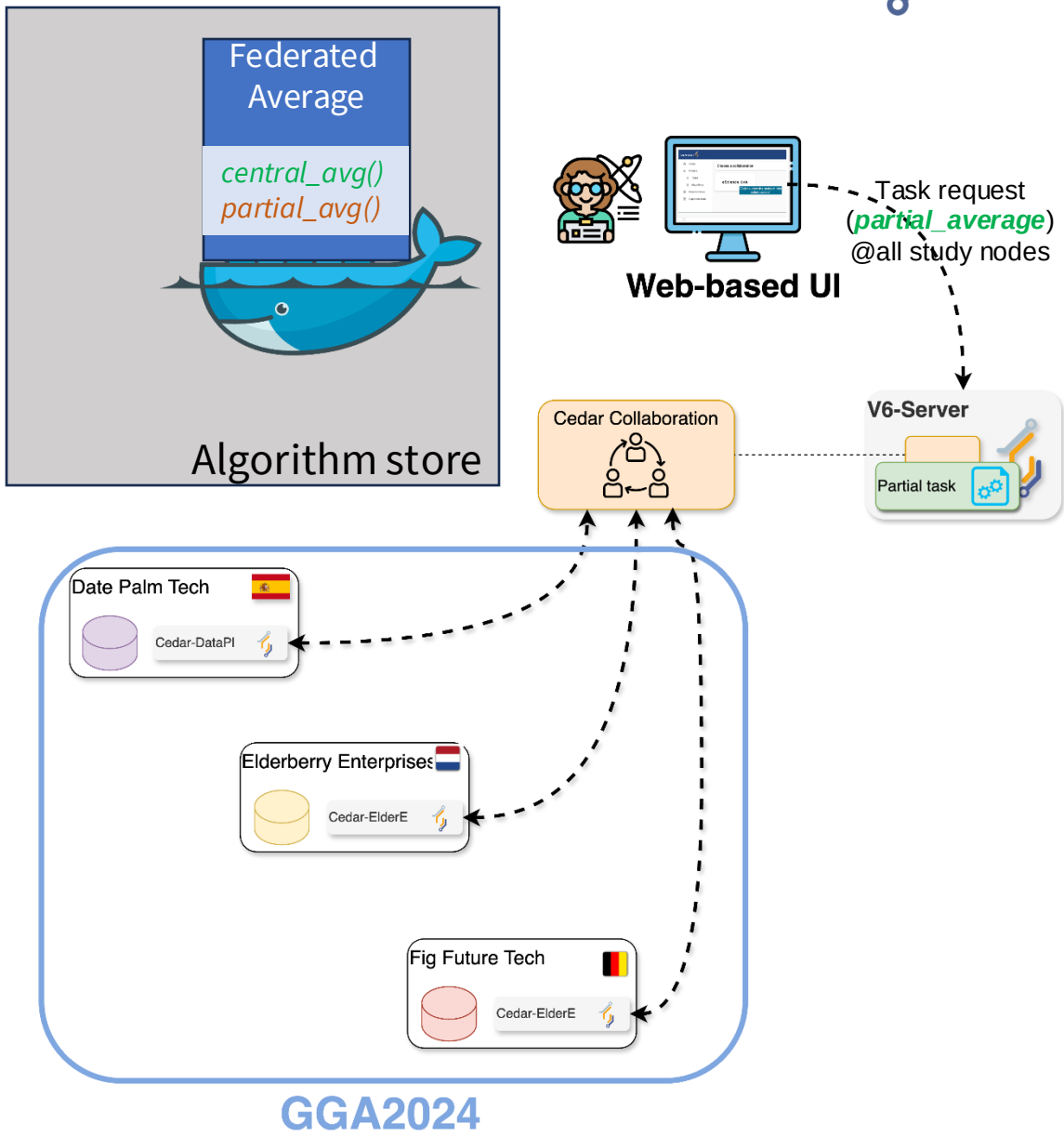


netherlands eScience center

KNL integraal kankercentrum Nederland

Running a PET: federated analysis

- Creating a new task:
  - You will run the ***partial_average*** function, of the ***Federated Average algorithm***, on **all** the nodes of a study (the one with no offline nodes).



Running a federated algorithm

VANTAGE

CHALLENGE

Create your first task!

- Use the *study* with **no** offline nodes.
- **Give your own name to the task so you can find it later.**
- Choose the Average algorithm.
- Select the ***partial_average*** function and all the organizations within the study.
- Choose the ***default*** database.
- Choose a numerical variable as the input.

The task you just requested should be listed with a 'pending' status. Once finished, download the JSON results and open them on a text editor.

- What does the content of these files mean? Why the results are formatted like that?

gender	age	height	weight	isOverweight	ageGroup
M	39	152	108	False	30 - 40
M	8	118	106	False	0 - 10
M	16	161	110	True	10 - 20
M	94	110	115	True	90 - 100
M	47	117	152	True	40 - 50
F	29	127	110	True	20 - 30
M	5	95	65	False	0 - 10
M	39	142	196	False	30 - 40
F	20	189	112	False	20 - 30
F	84	145	116	False	80 - 90

HINT
The data on 'default' databases on all the nodes looks like this ☝.

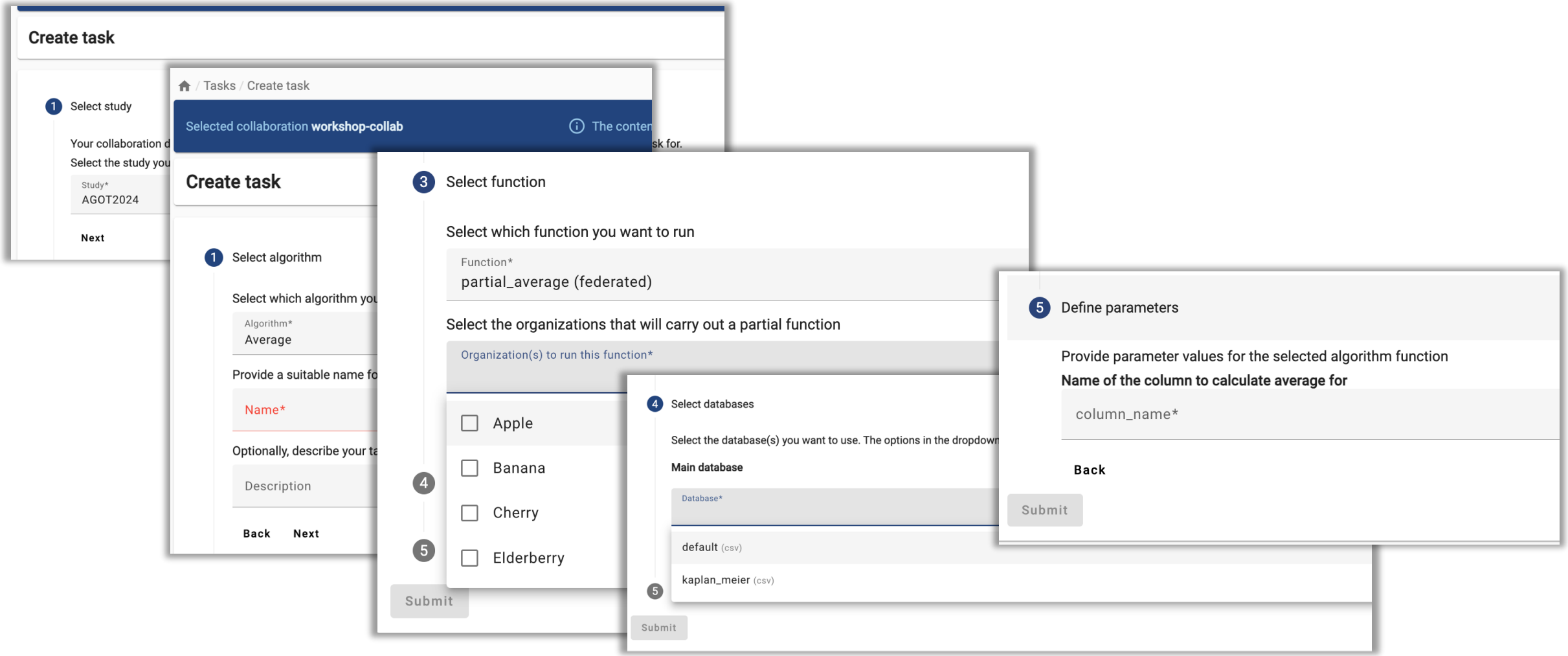
netherlands eScience center

integraal kankercentrum Nederland

Running a federated algorithm

VANTAGE

SOLUTION



netherlands eScience center

KNL integraal kankercentrum Nederland

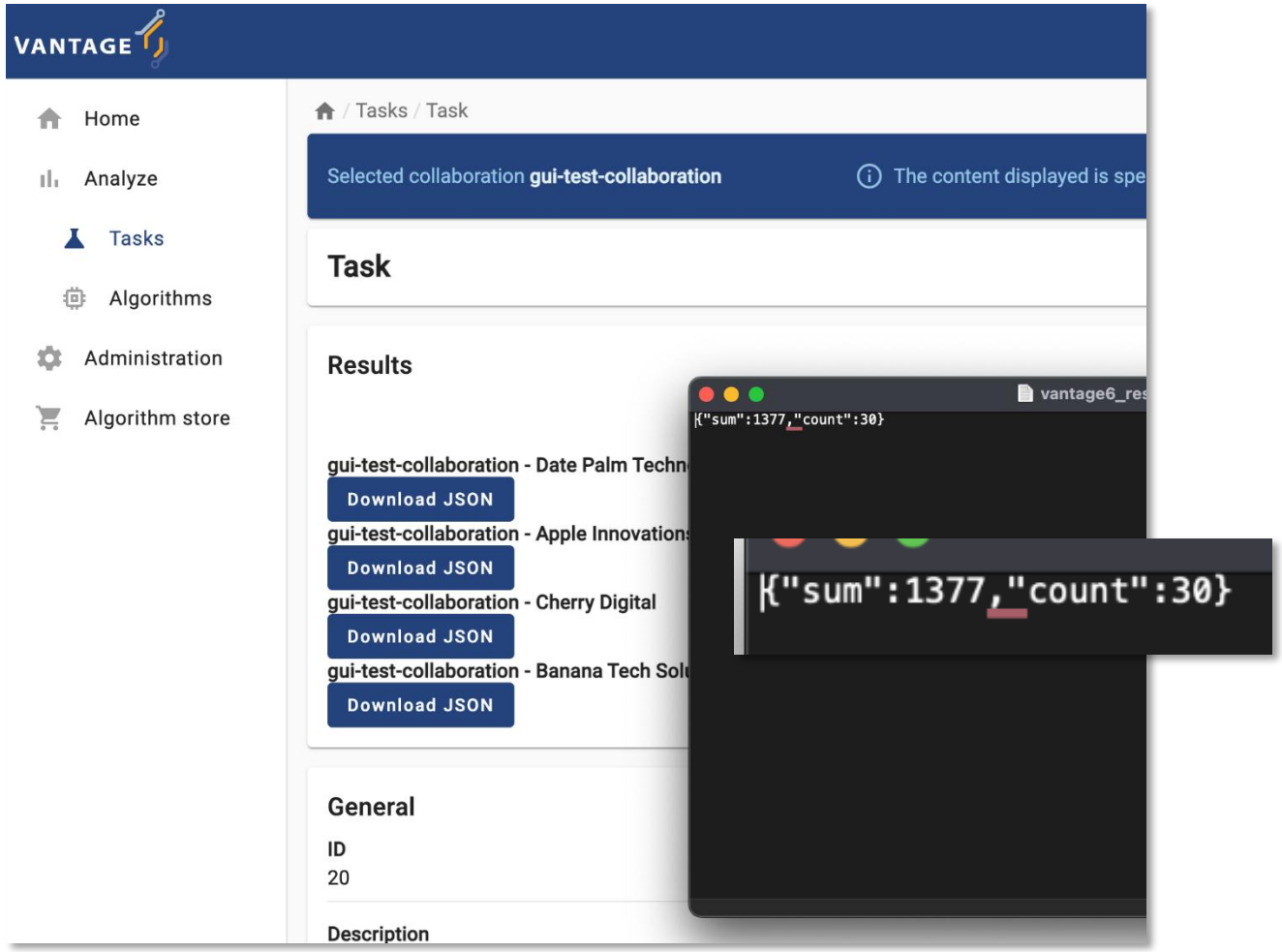
Getting familiar with the UI

VANTAGE

SOLUTION

The ***partial_average*** returns the two values needed by the central function of the ‘federated average’ algorithm, as described in Chapter 2: the number of records within the database, and their sum.

The algorithm is ‘encoding’ these values as a JSON document, so they can used later (e.g., by another function or program).

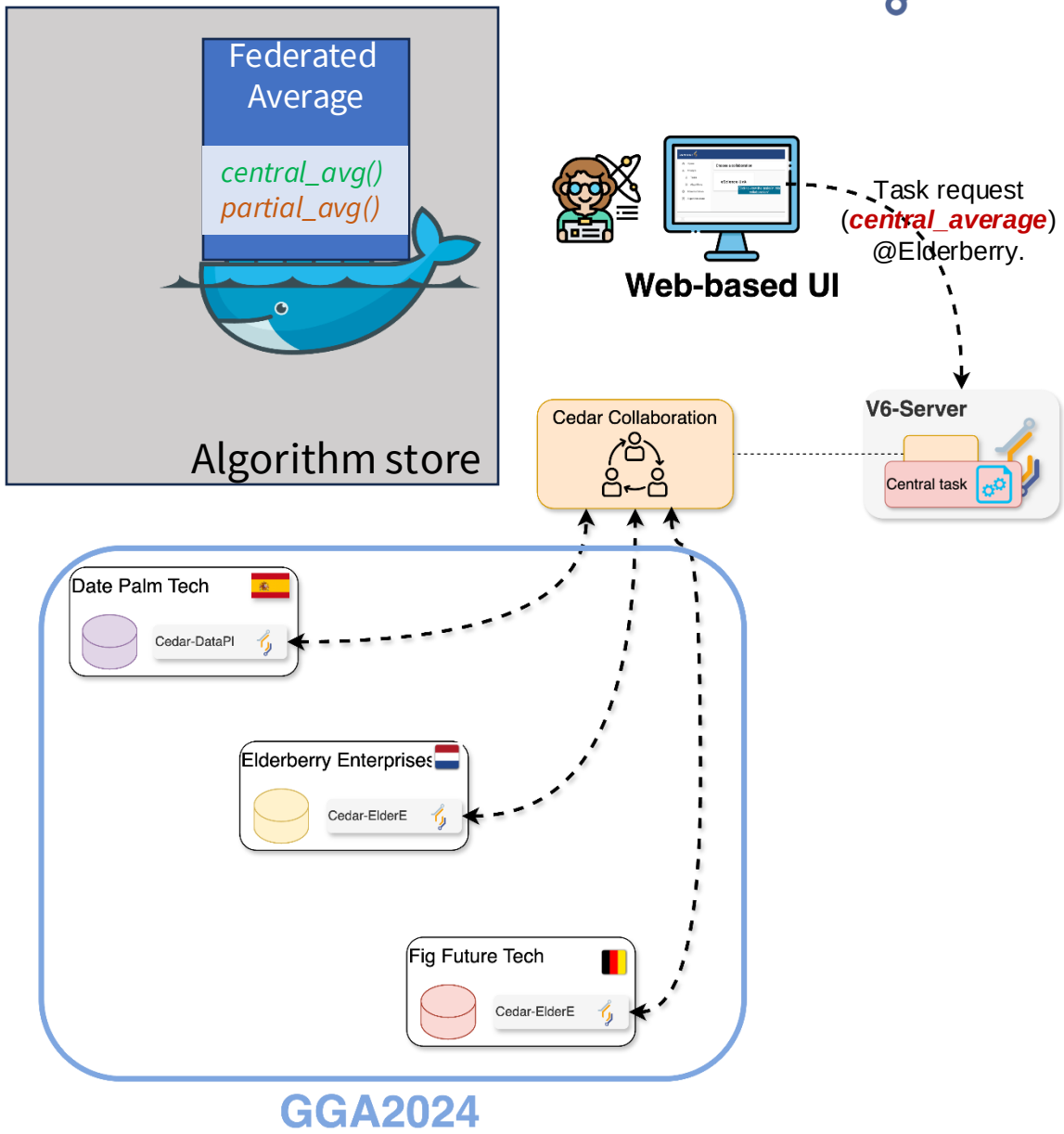


netherlands eScience center

integraal kankercentrum Nederland

Running a PET: federated analysis

- Let's create yet another task.
  - This time you will run the ***central_average*** function, of the ***Federated Average algorithm***, on **one** of the study nodes (once again, on the study with **no** offline nodes).



Running a federated algorithm

VANTAGE

CHALLENGE

Let's see what the central function does under the hood!

- Use the study with **no** offline nodes.
- Choose the **Average algorithm**.
- Select the ***central_average*** function and one of the organizations within the study.
- Choose the default database.
- Choose a numerical variable as the input.

Before submitting the request: why does the V6 UI let you select only a single organization for the central function?

Keep an eye on the Tasks section, and see how the processes are created:

- Why does one of the nodes show up as a Main process and also as a Child one?

HINT

As soon your task is created, go to the task list and open it. See how 'child processes' are created.

netherlands eScience center

integraal kankercentrum Nederland

Algorithm pseudo-code

VANTAGE

```
FUNCTION partial_average(dataset, column_name)

1  INFO("Extracting column", column_name)
2  column_values = dataset[column_name]
3
4  INFO("Computing partials")
5  local_sum = SUM(column_values)
6  local_count = SIZE(column_values)
7
8  RETURN {"sum": local_sum, "count": local_count}

END FUNCTION
```

- Why does one of the nodes show up as a Main process and also as a Child one?

Actual python code?
https://bit.ly/v6-federated-avg

```
FUNCTION central_average(client, column_name)

1   study_orgs = client.study.list()
2
3
4   INFO("Requesting partial computation")
5   task = client.task.create(
6     function = "partial_average"
7     column = column_name
8     organizations = study_orgs
9   )
10
11  INFO("Waiting for results")
12  results = client.wait_for_results(task)
13
14  global_sum = 0
15  global_count = 0
16
17  FOR EACH output IN results
18    global_sum = global_sum + output.sum
19    global_count = global_count + output.count
20
21  average = global_sum/global_count
22
23  RETURN {"average": average}

END FUNCTION
```

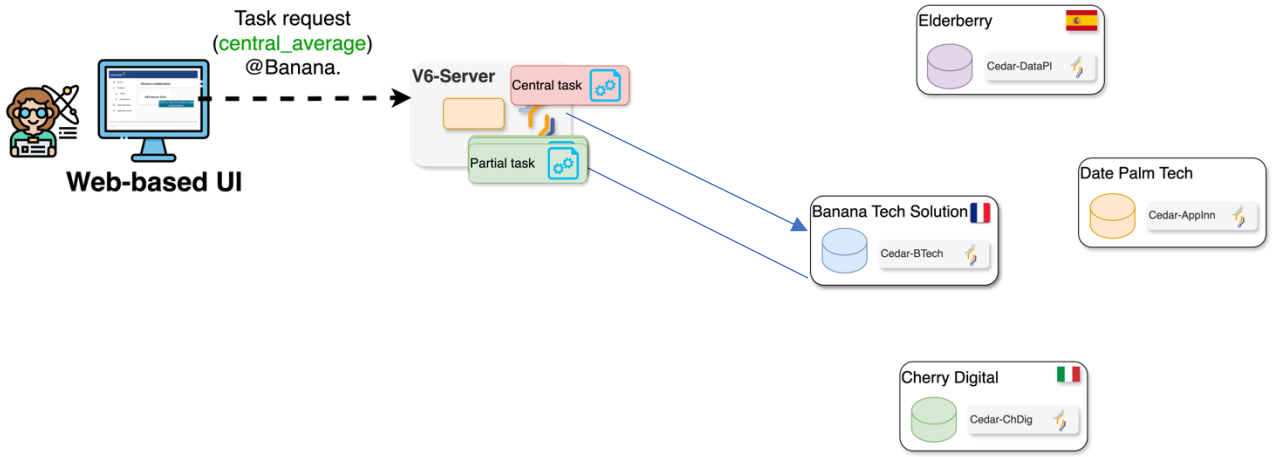
Getting familiar with the UI

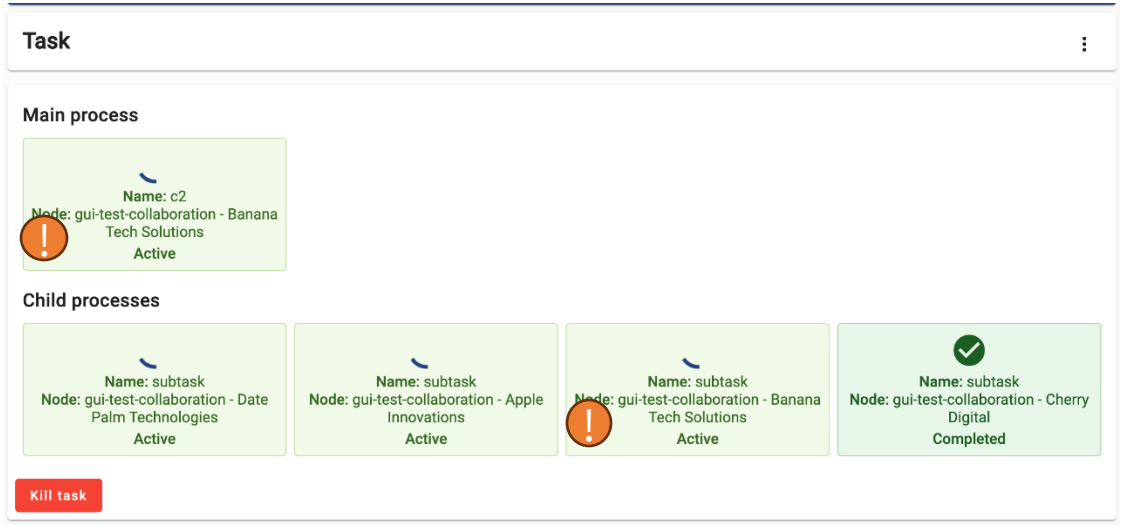
VANTAGE

SOLUTION

The **Task** with the **central_average** function, when executed by a node, requests **all** the nodes within the **Study** to run the **partial_average** one. This particular node is also part of the study, so in the end it gets his own request!

The **central_average** was designed to just consolidate the results of the **partial_average**, sent by the other nodes. Hence, it doesn't need direct access any dataset.





netherlands eScience center

integraal kankercentrum Nederland

Double-checking core concepts…

VANTAGE

```
FUNCTION partial_average(dataset, column_name)

1  INFO("Extracting column", column_name)
2  column_values = dataset[column_name]
3
4  INFO("Computing partials")
5  local_sum = SUM(column_values)
6  local_count = SIZE(column_values)
7
8  RETURN {"sum": local_sum, "count": local_count}

END FUNCTION
```

- Why does the ***central_average*** function, unlike ***partial_average***, **not** get any ***dataset*** as an input?
- Why does the ***partial_average*** function, unlike ***central_average***, **not** get any ***client*** as an input?

```
FUNCTION central_average(client, column_name)

1    study_orgs = client.study.list()
2
3
4  INFO("Requesting partial computation")
5  task = client.task.create(
6      function = "partial_average"
7      column = column_name
8      organizations = study_orgs
9  )
10
11 INFO("Waiting for results")
12 results = client.wait_for_results(task)
13
14 global_sum = 0
15 global_count = 0
16
17 FOR EACH output IN results
18   global_sum = global_sum + output.sum
19   global_count = global_count + output.count
20
21 average = global_sum/global_count
22
23 RETURN {"average": average}

END FUNCTION
```

netherlands eScience center

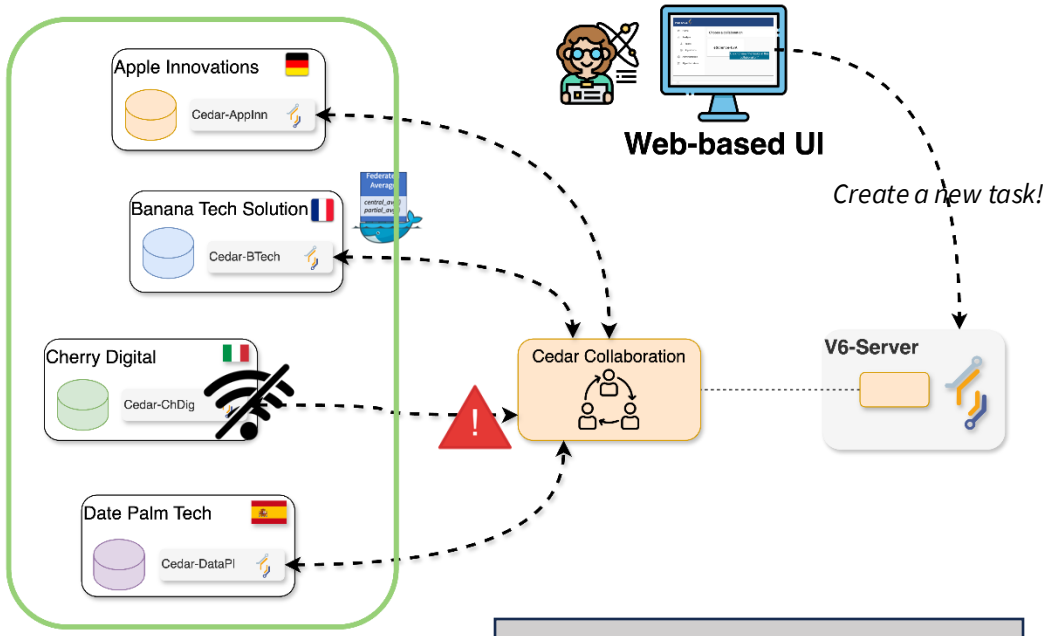
KNL integraal kankercentrum Nederland

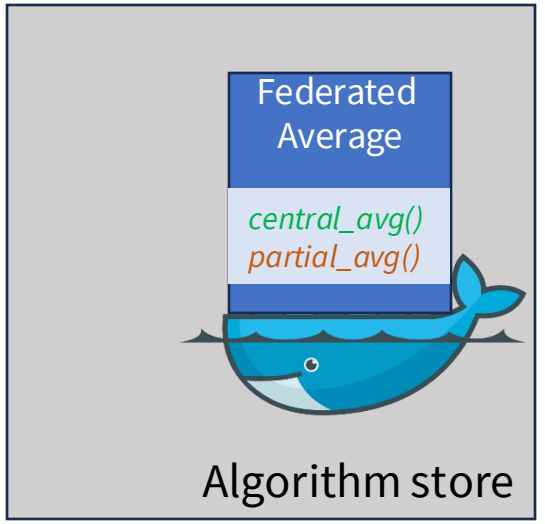
Running a PET: federated analysis

VANTAGE

Problems on a node?

- What if…
  - You run the ***central_average*** function (***Federated Average algorithm)***, on the study that has an **offline** node?





netherlands eScience center

integraal kankercentrum Nederland

Running a federated algorithm

VANTAGE

CHALLENGE

handling problems through the UI!

- Use the study that has **an offline node**.
- Choose the **Average algorithm**.
- Select the ***central_average*** function and an **online** organizations within the study.
- Choose the *default* database.
- Choose a numerical variable as the input.

Before starting the task try to speculate: what is going to happen?

```
FUNCTION central_average(client, column_name)

    study_orgs = client.study.list()


  INFO("Requesting partial computation")
  task = client.task.create(
      function = "partial_average"
      column = column_name
      organizations = study_orgs
  )

  INFO("Waiting for results")
  results = client.wait_for_results(task)

  global_sum = 0
  global_count = 0

  FOR EACH output IN results
      global_sum = global_sum + output.sum
      global_count = global_count + output.count

  average = global_sum/global_count

  RETURN {"average": average}

END FUNCTION
```

netherlands eScience center

integraal kankercentrum Nederland

Getting familiar with the UI

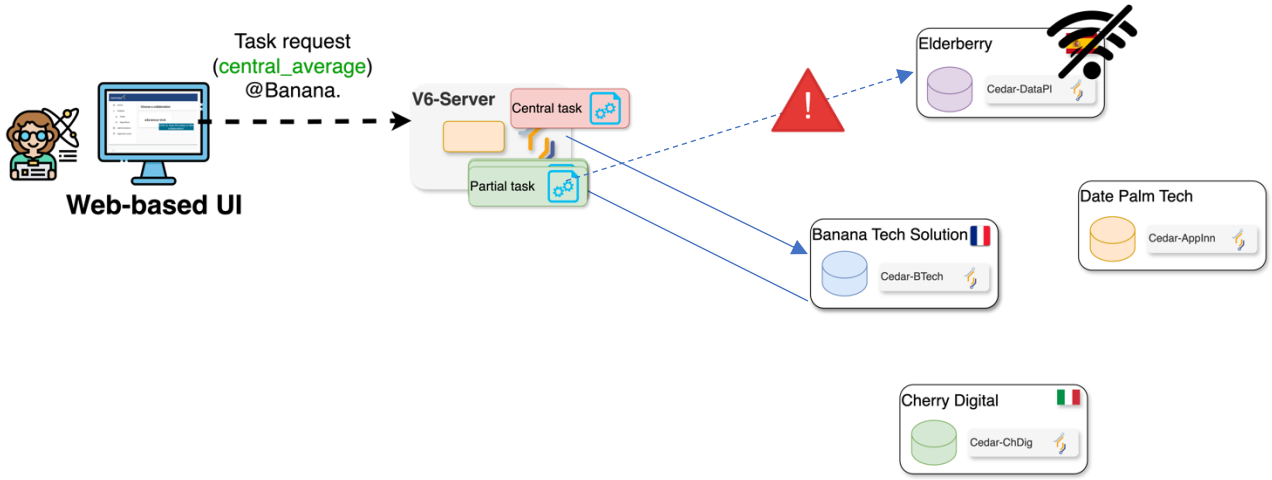
VANTAGE

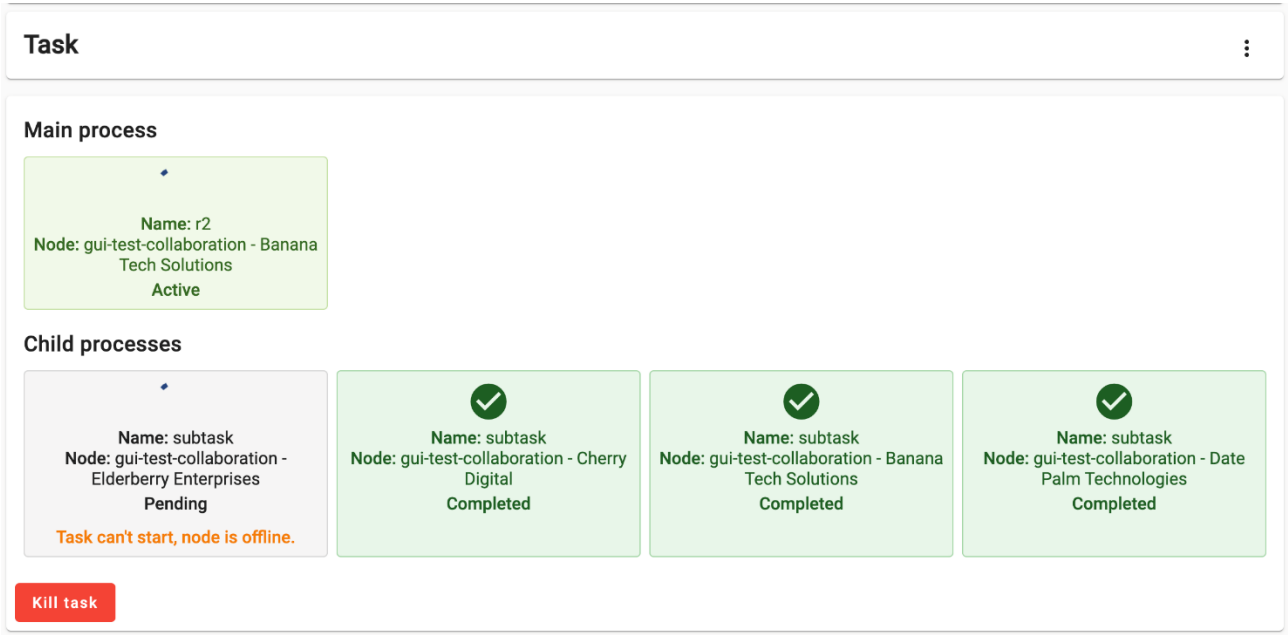
SOLUTION

The algorithm didn’t crash.

The Central task requests all the nodes in the study to run the ‘partial’ function. As the server is unable to transfer this request to the offline node, this child process is kept on hold, until the node is back online.

Consequently, the Main process is also kept on hold, and the process stays with an ‘Active’ status indefinitely (or until the node is back online).





netherlands eScience center

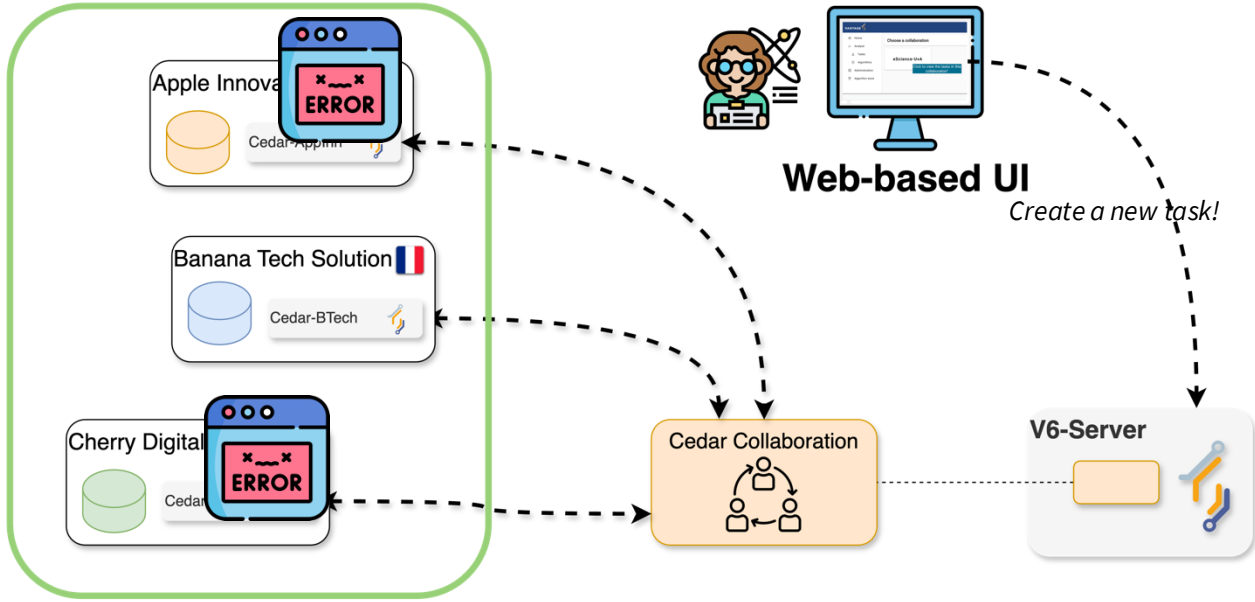
integraal kankercentrum Nederland

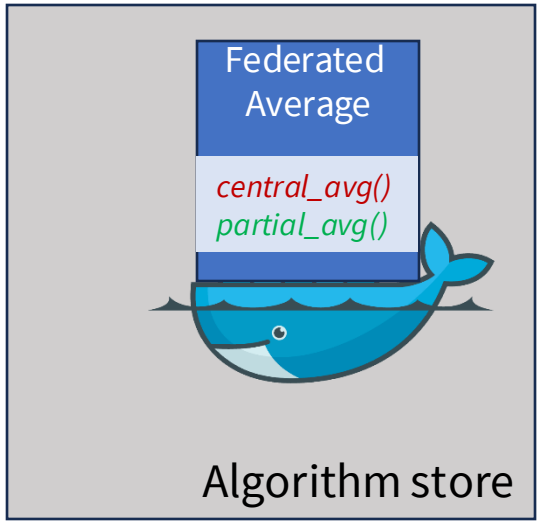
Running a PET: federated analysis

VANTAGE

Problems at the code level?

- What if…
  - You run the ***central_average*** function (***Federated Average algorithm)***, *and later on* a node gets a runtime error? (e.g., a code bug, unexpected input, etc.)





netherlands eScience center

integraal kankercentrum Nederland

Running a federated algorithm

VANTAGE

CHALLENGE

Handling problems through the UI!

- Use the study that has **no** offline nodes.
- Choose the **Average algorithm**.
- Select the ***central_average*** function and one of the organizations within the study.
- Choose the default database.
- Choose the **ageGroup** variable as the input.

Look at the error logs. What was the cause of the failure of the ***partial_average***? What about the ***central_average***?

gender	age	height	weight	isOverweight	ageGroup
M	39	152	108	False	30 - 40
M	8	118	106	False	0 - 10
M	16	161	110	True	10 - 20
M	94	110	115	True	90 - 100
M	47	117	152	True	40 - 50
F	29	127	110	True	20 - 30
M	5	95	65	False	0 - 10
M	39	142	196	False	30 - 40
F	20	189	112	False	20 - 30
F	84	145	116	False	80 - 90

HINT

The error logs contain the error messages generated by the Python interpreter while running the algorithm functions. Look for messages that hint the root cause of the problem.

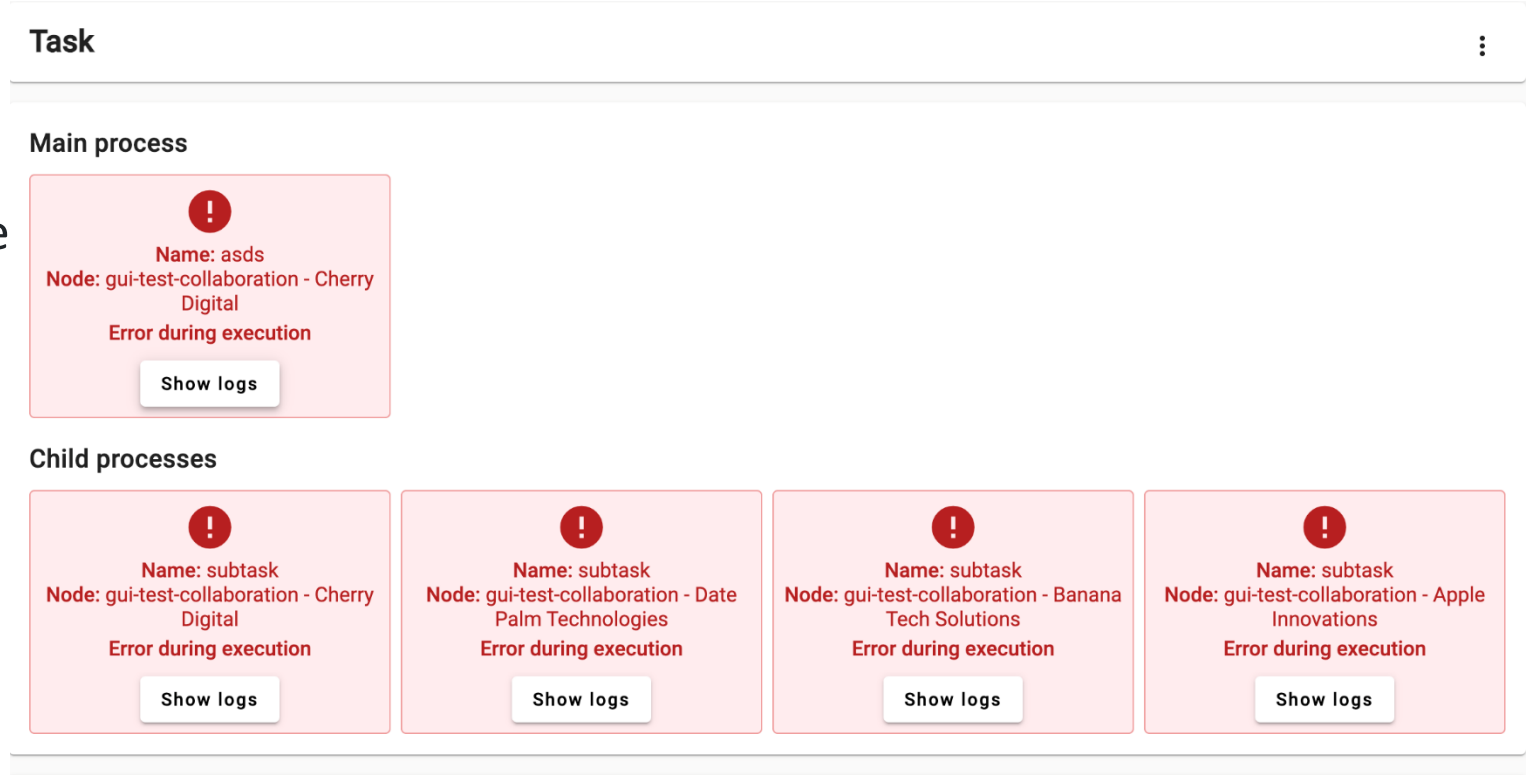
Getting familiar with the UI

VANTAGE

SOLUTION

1. This time, the ***partial_average*** function on all the data nodes crashes, which lead to the failure of the Main process (***central_average***).

2. The ***partial_average*** crashed while trying to parse the input as a number.
   The ***central_average*** crashed due to a division-by-zero.



Play around with other algorithms

CHALLENGE

Try to see if you can run a different algorithm than the average we have used so far:

Kaplan–Meier estimator

HINT
This time, choose the *kaplan_meier (csv)* database!