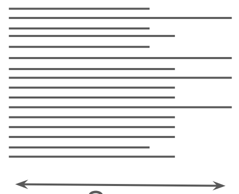
index logan with kmindex

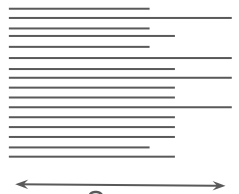
Pipeline

Overview

n accessions.



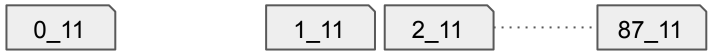
n accessions.



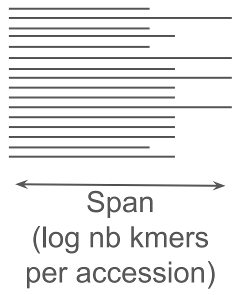
(log nb kmers per accession)

For each span (one bloom filter size)

1. Create subGroups
2048 accessions per subgroup (except for largest)

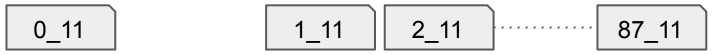


n accessions.

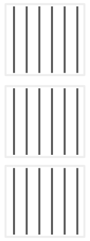


For each span
one bloom filter size

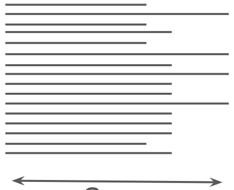
1. Create subGroups
2048 accessions per group (except for largest)



2. Computes:
* repartition
* partitioned BFs



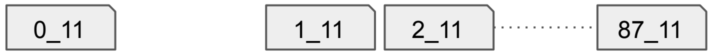
n accessions.



Span
(log nb kmers per accession)

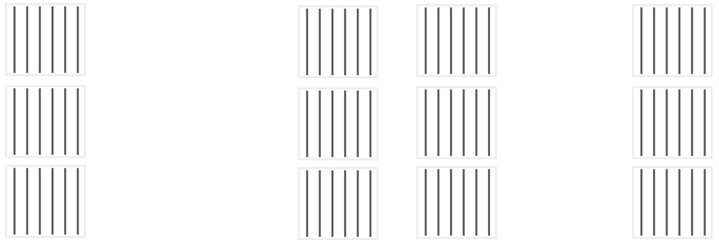
For each span
one bloom filter size

1. Create subGroups
2048 accessions per group (except for largest)

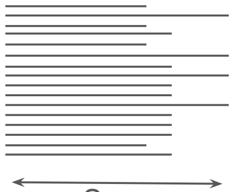


2. Computes:
* repartition
* partitioned BFs

3. Computes:
* partitioned BFs
using repartition



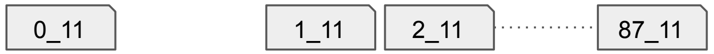
n accessions.



For each span

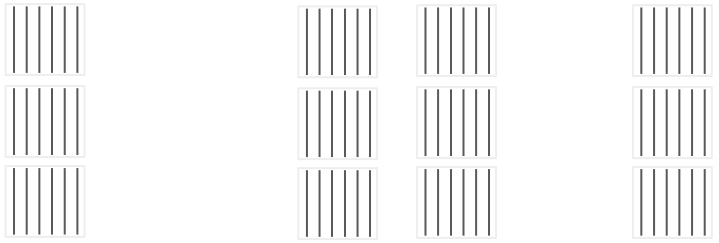
one bloom filter size

1. Create subGroups
2048 accessions per group (except for largest)

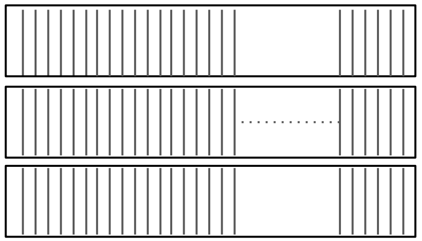


2. Computes:
* repartition
* partitioned BFs

3. Computes:
* partitioned BFs
using repartition



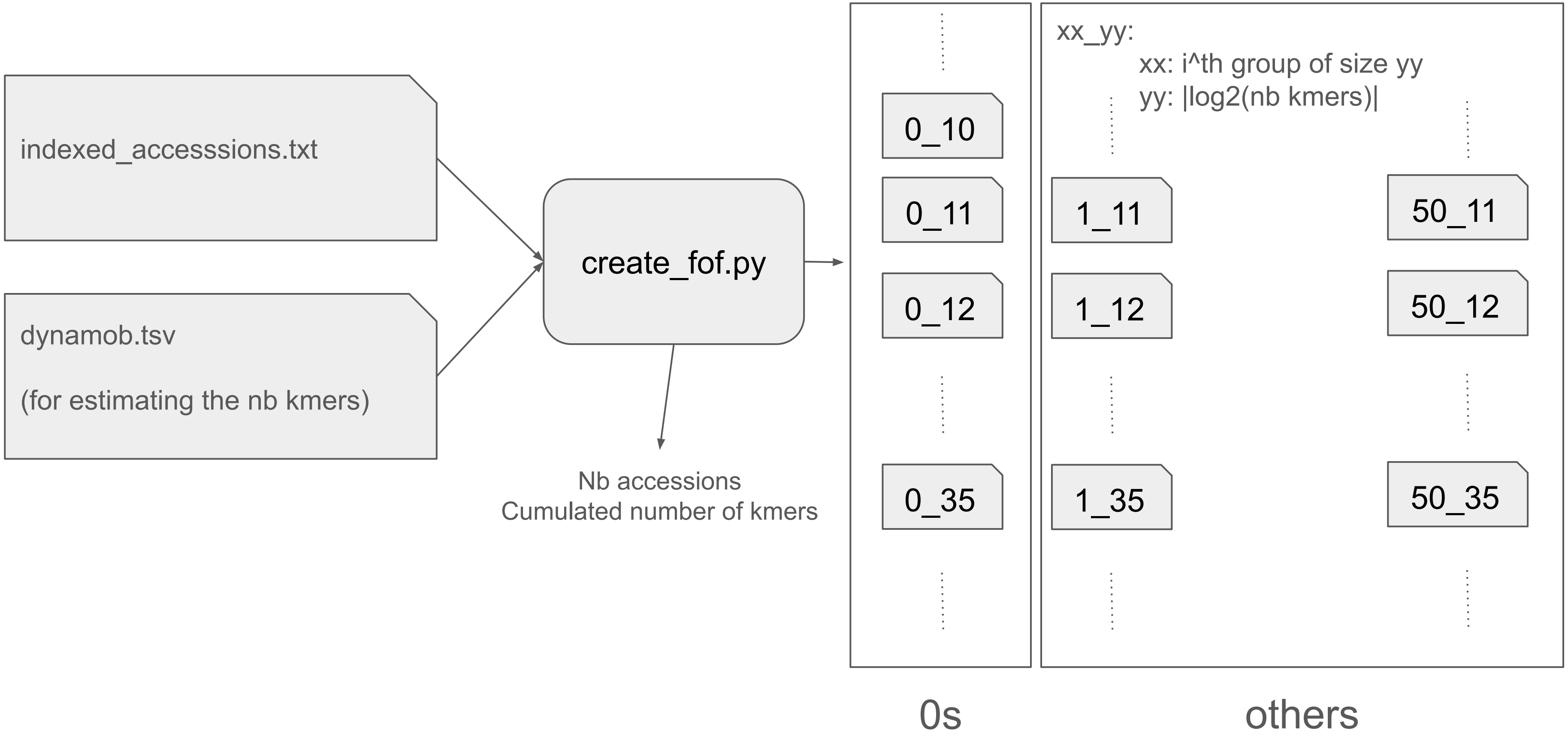
4. Merge



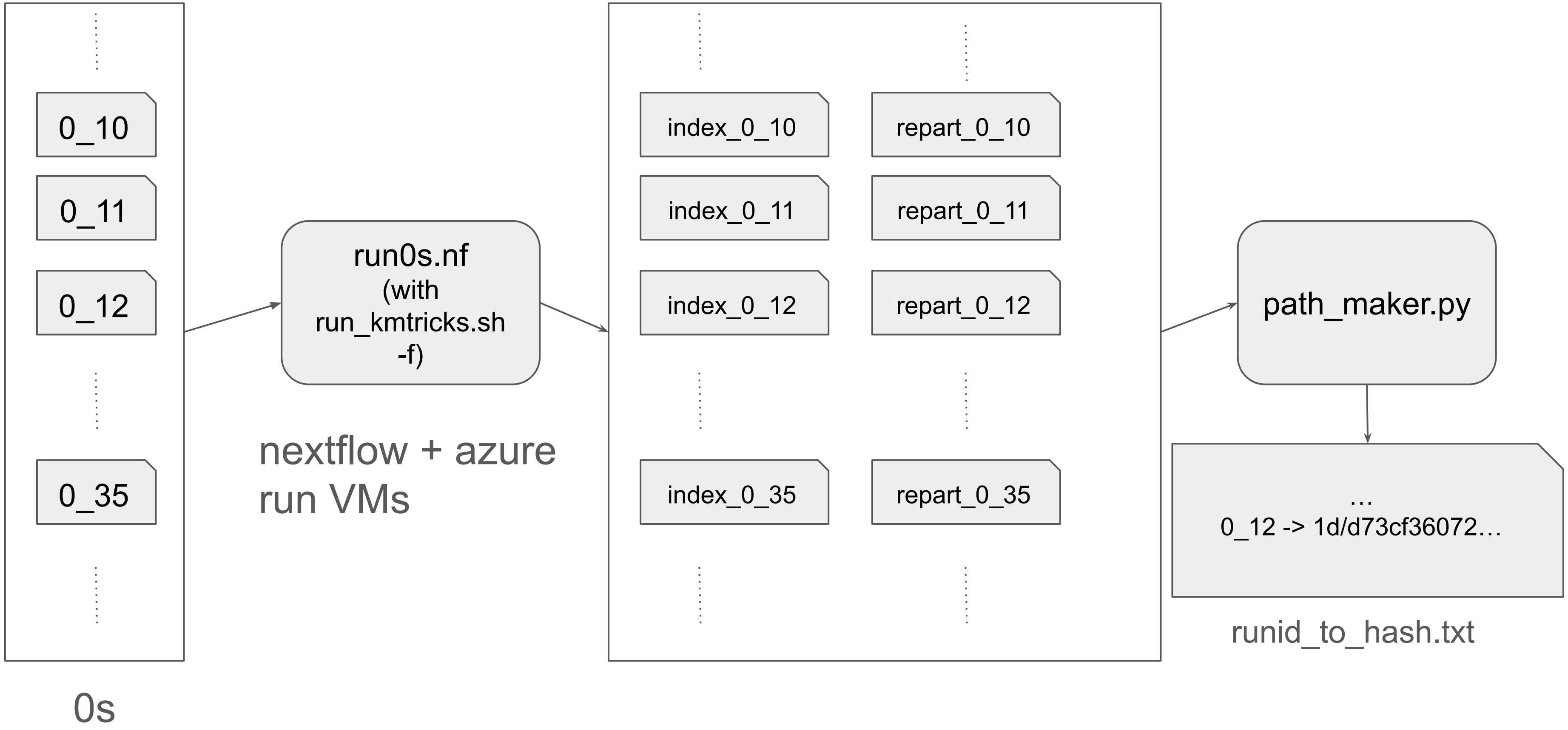
One queryable index
one group x one span

Some details

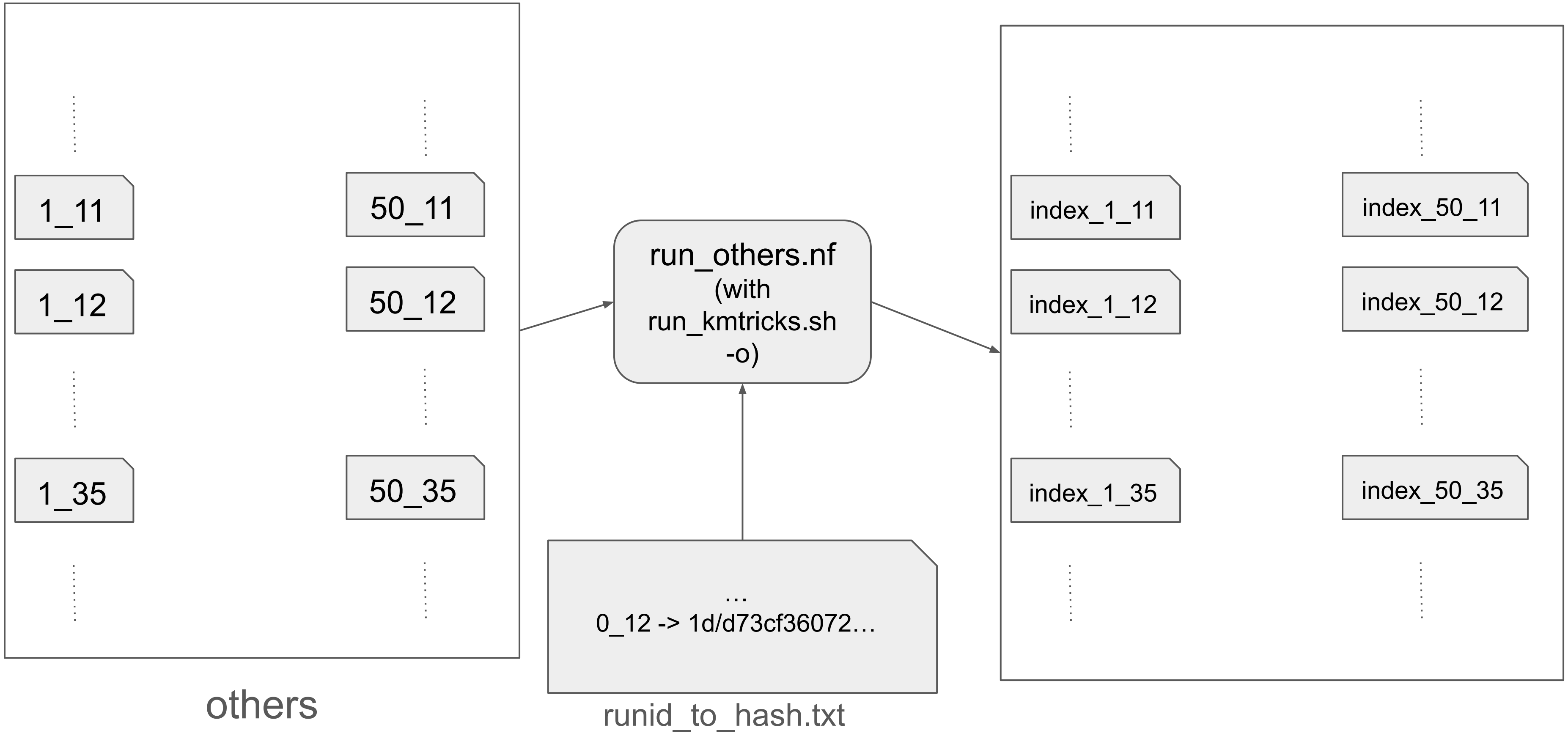
From groups to FOF (`create_fof.py`)



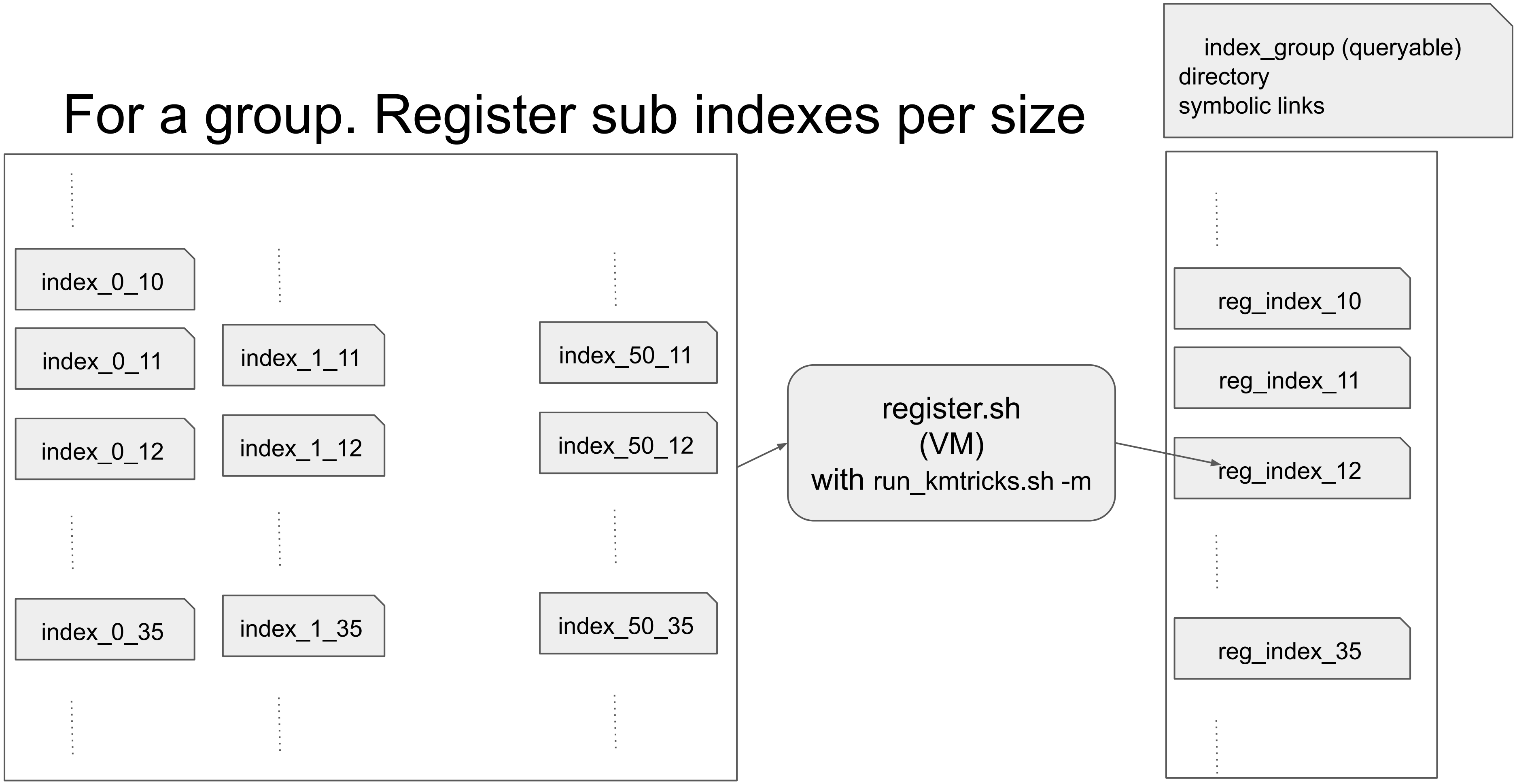
For a group. From Os to sub indexes



For a group. From others to sub indexes

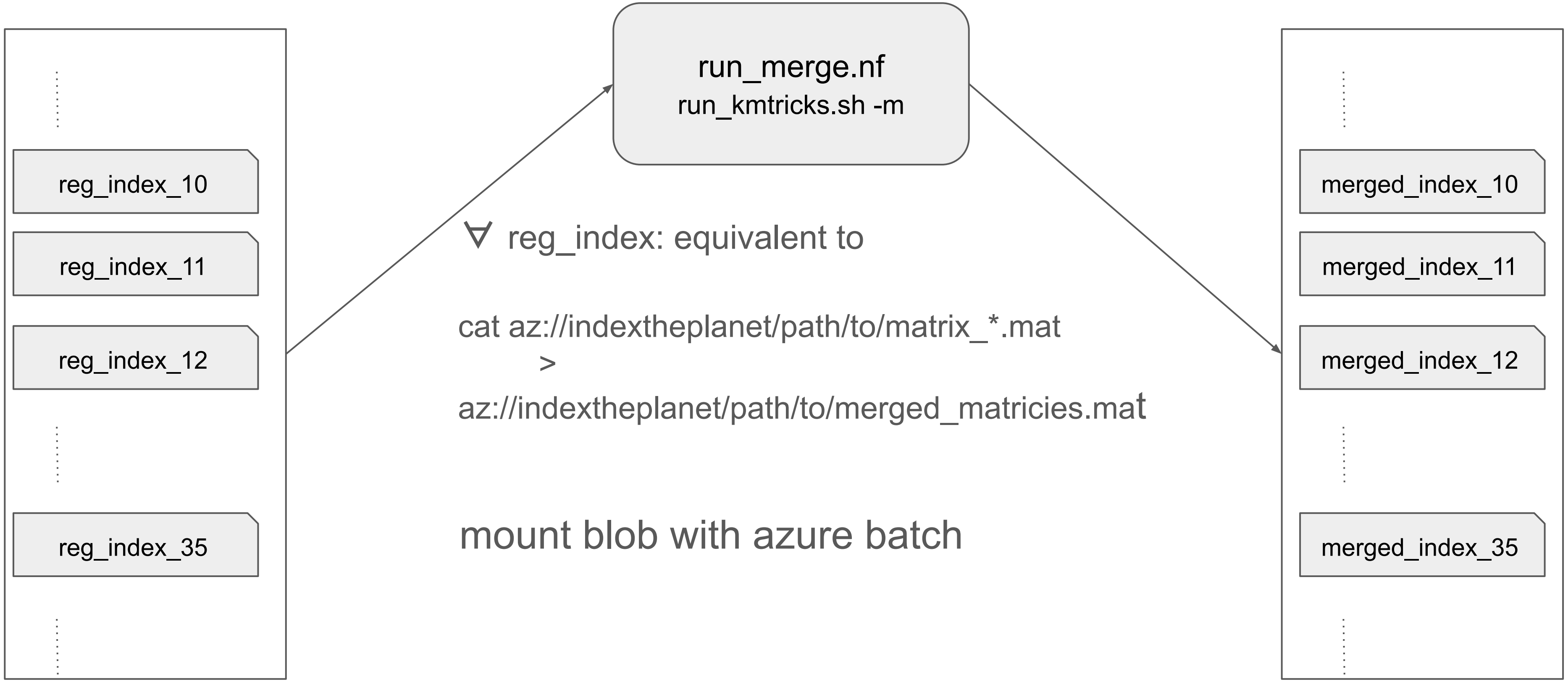


For a group. Register sub indexes per size



For a group. Merge sub indexes per size

One merged index per size



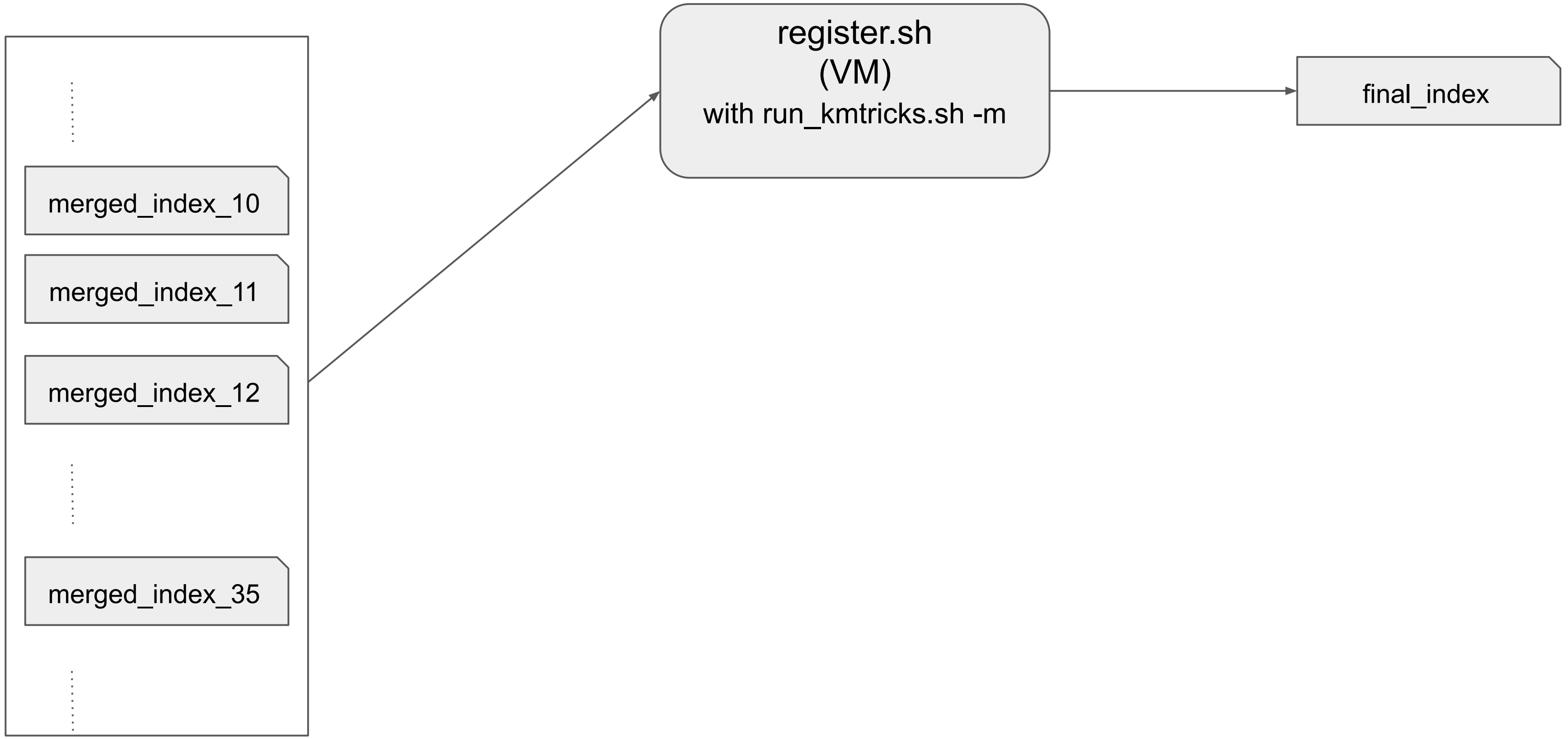
∀ reg_index: equivalent to

```
cat az://indextheplanet/path/to/matrix_*.mat
        >
az://indextheplanet/path/to/merged_matricies.mat
```

mount blob with azure batch

For a group. Final register

One merged index per size



For a group, clean redundant matrices

During the merging

- we do not remove matrices that were merged.
- we test if the merging was correct (validate_merge.py)

After merging and validation we clean the old matrices that were merged:

- launch_clean.sh