Deduplicating Cloud Functions - Demo 3

Members:

Beliz Kaleli

Vikash Sahu

Paritosh Shirodkar

Asutosh Patra

Mentor:

Shripad Nadgowda

BOSTON UNIVERSITY

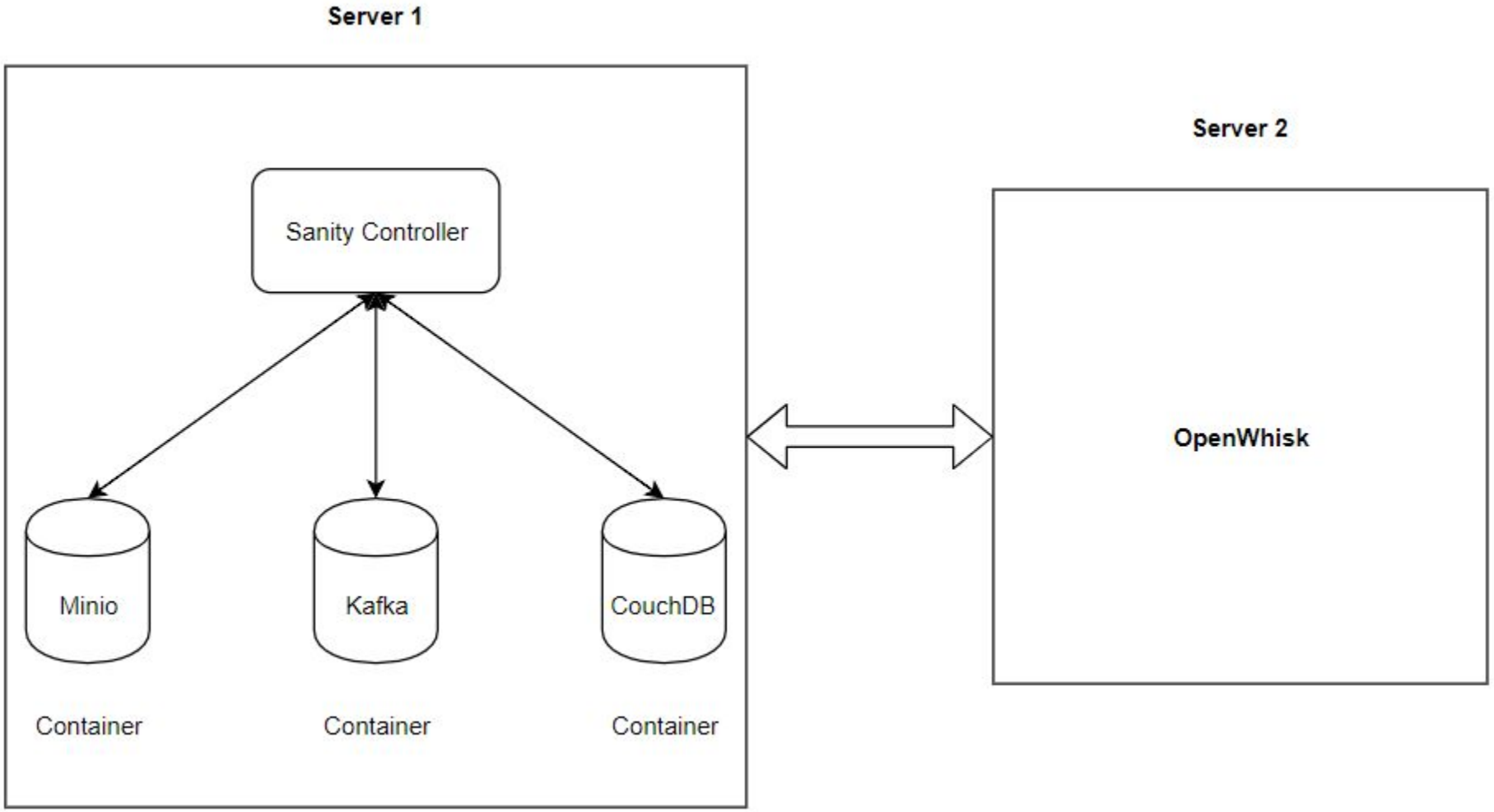
Previously in De-duplicating...

- Earlier, all the components (Minio, Kafka, CouchDB & OpenWhisk) were executing independently in local machines
- Interaction among the components was not done
- The feedback from the last sprint was to deploy the whole framework into cloud and execute

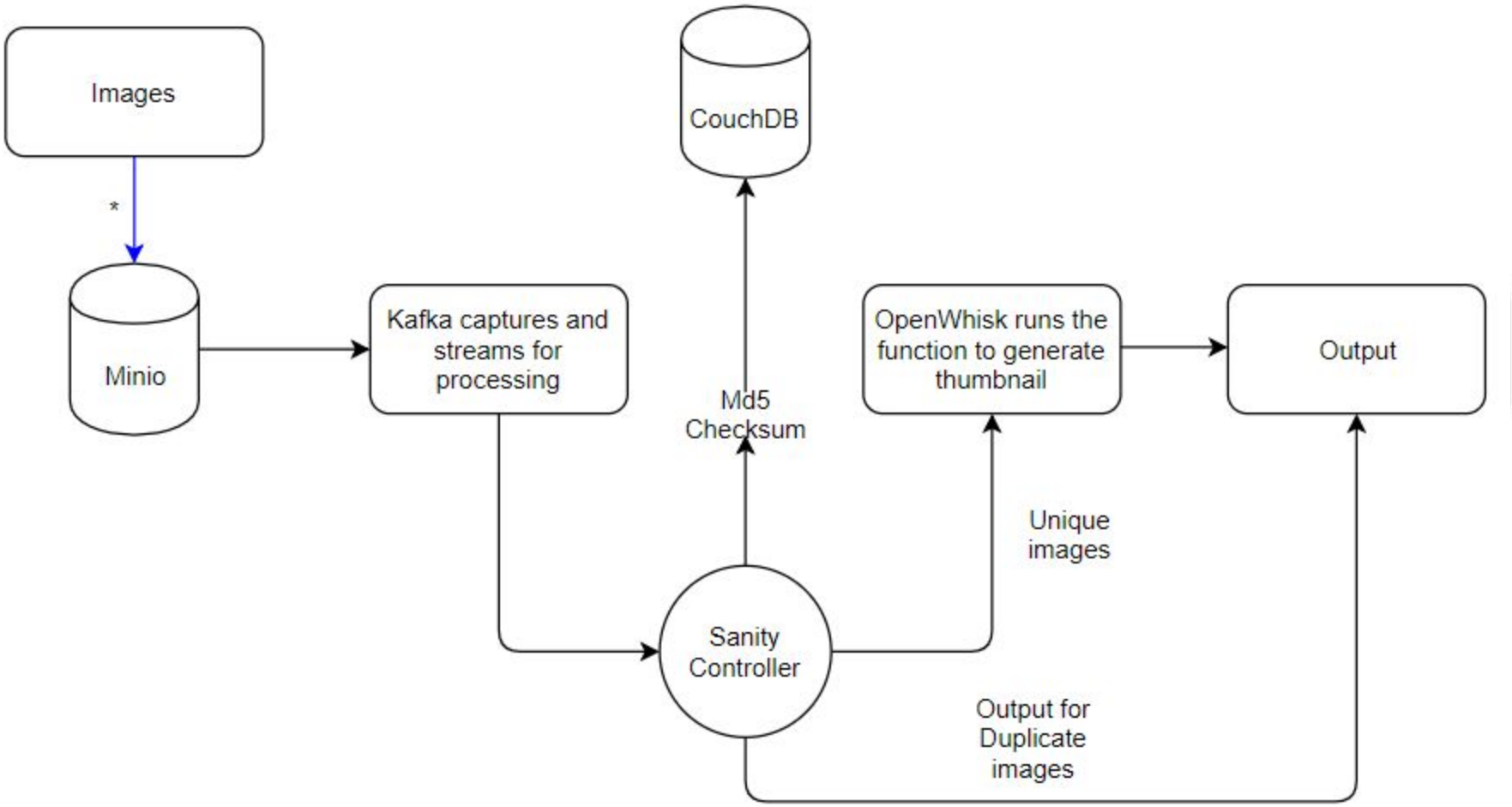
What did we achieve in this sprint?

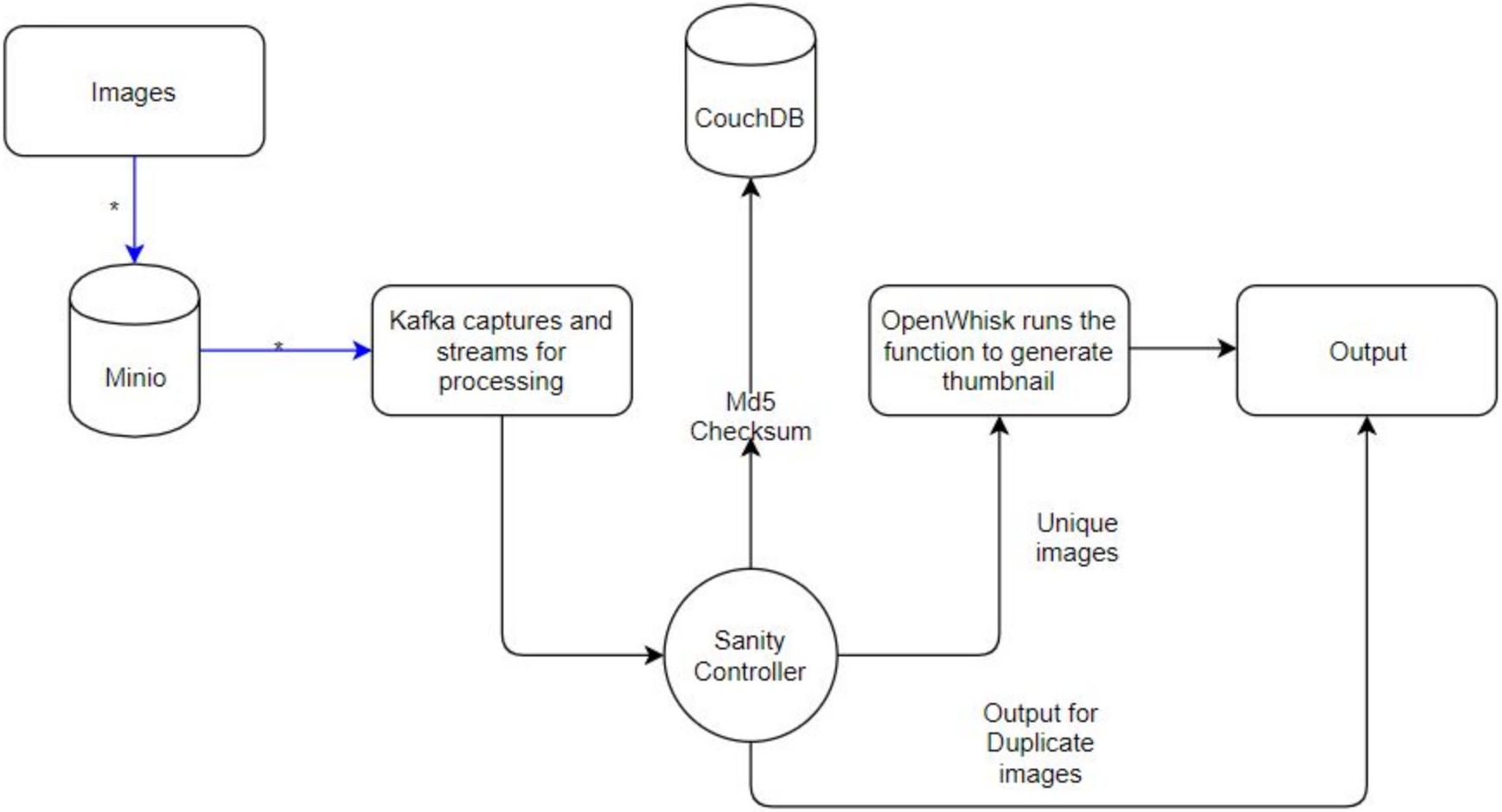
- Deployed all the individual components in the cloud
- Built the sanity framework to avoid deduplication of cloud function
- Successfully implemented **deduplication** of cloud function for **Image Thumbnail Use Case**

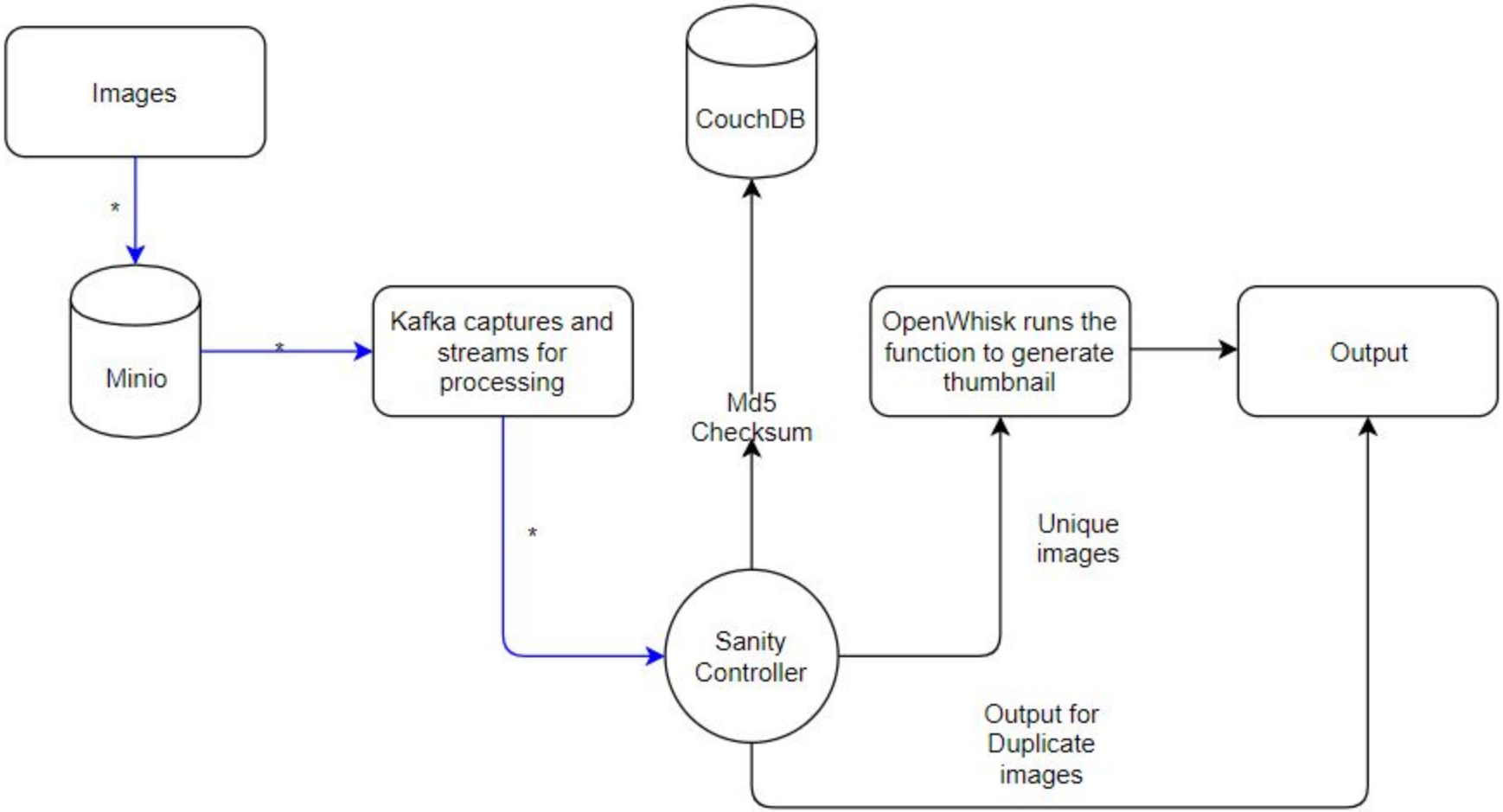
Now, Simplified working model in IBM Cloud

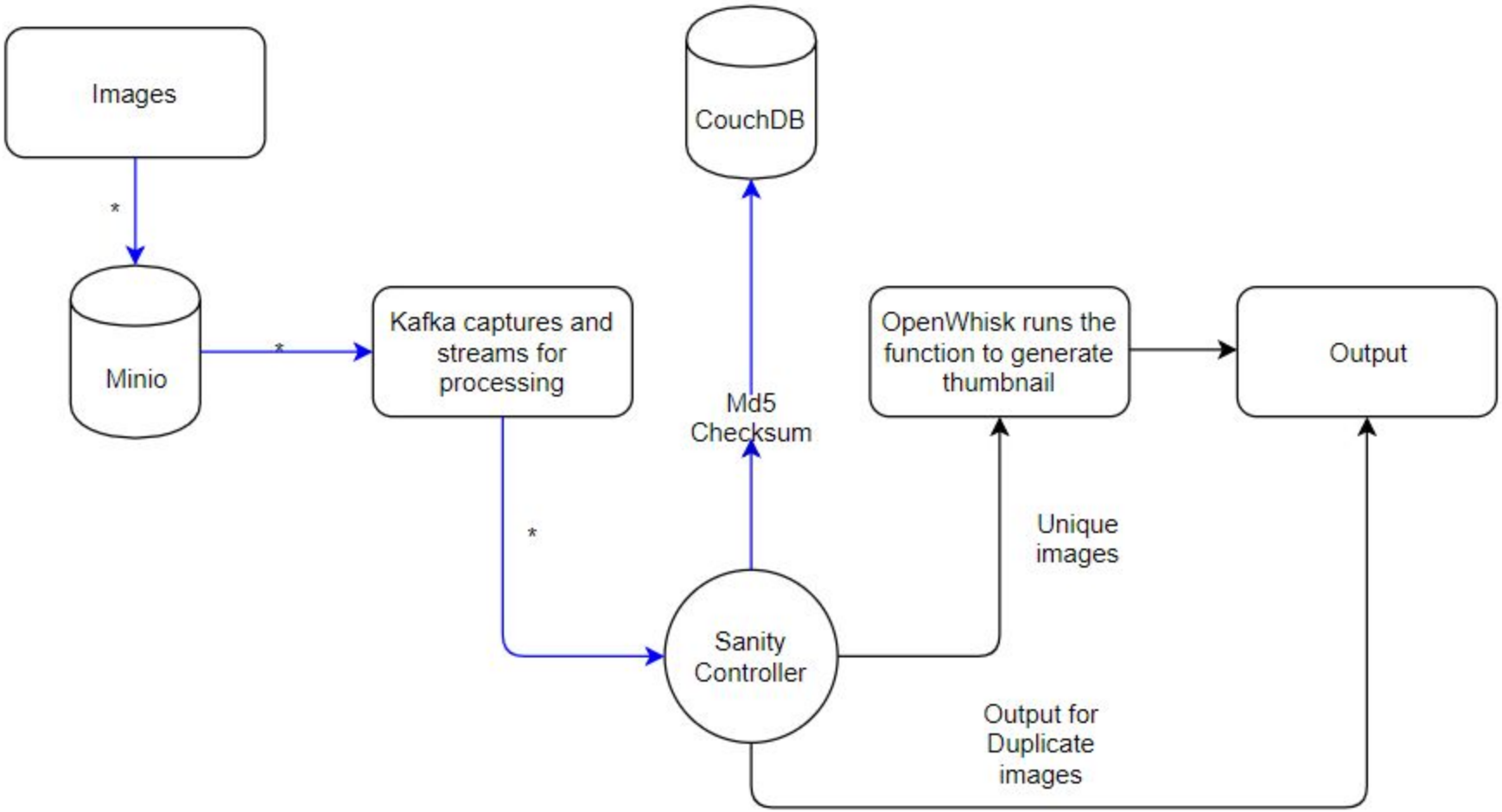


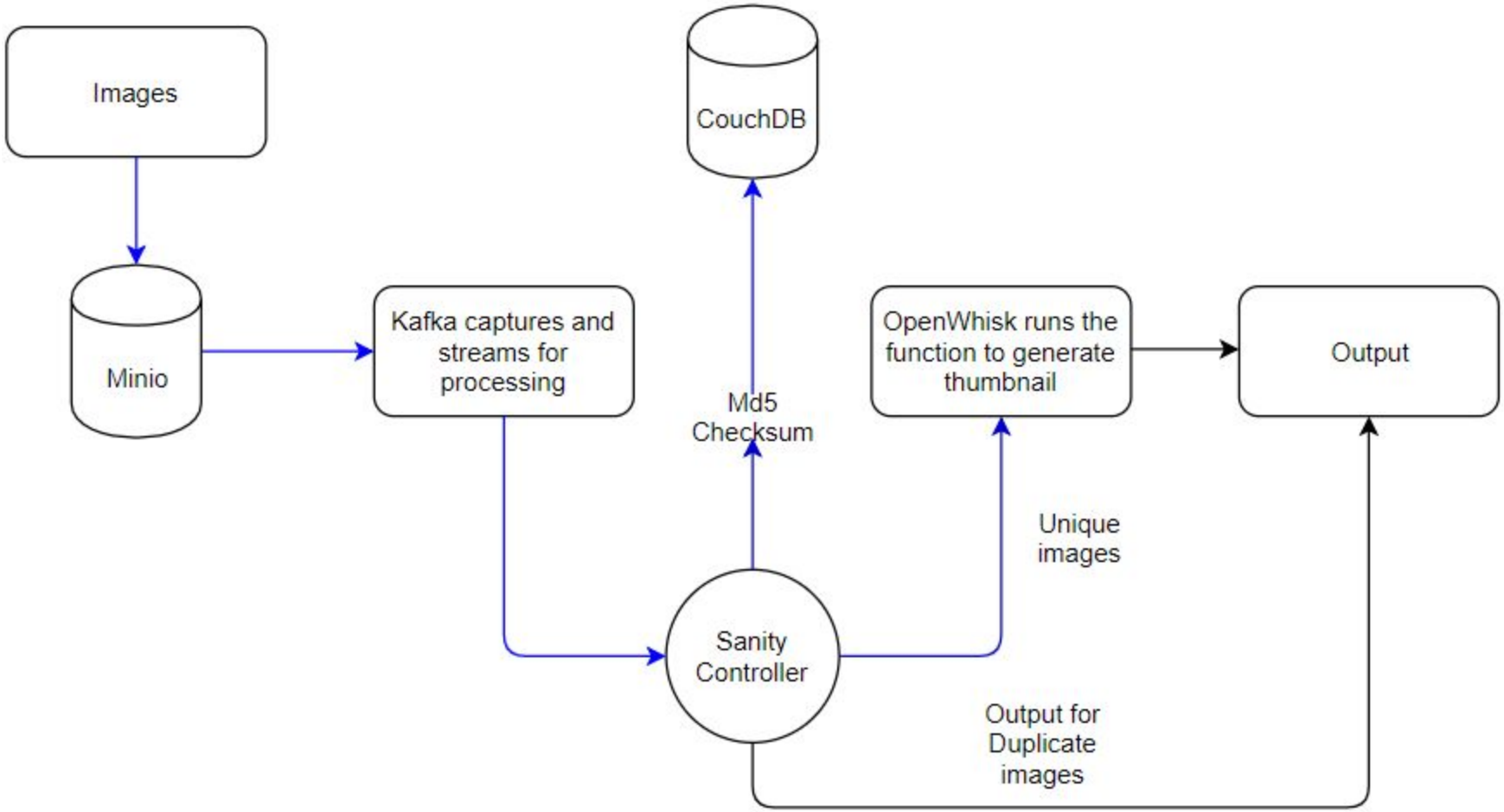
Architecture Diagram

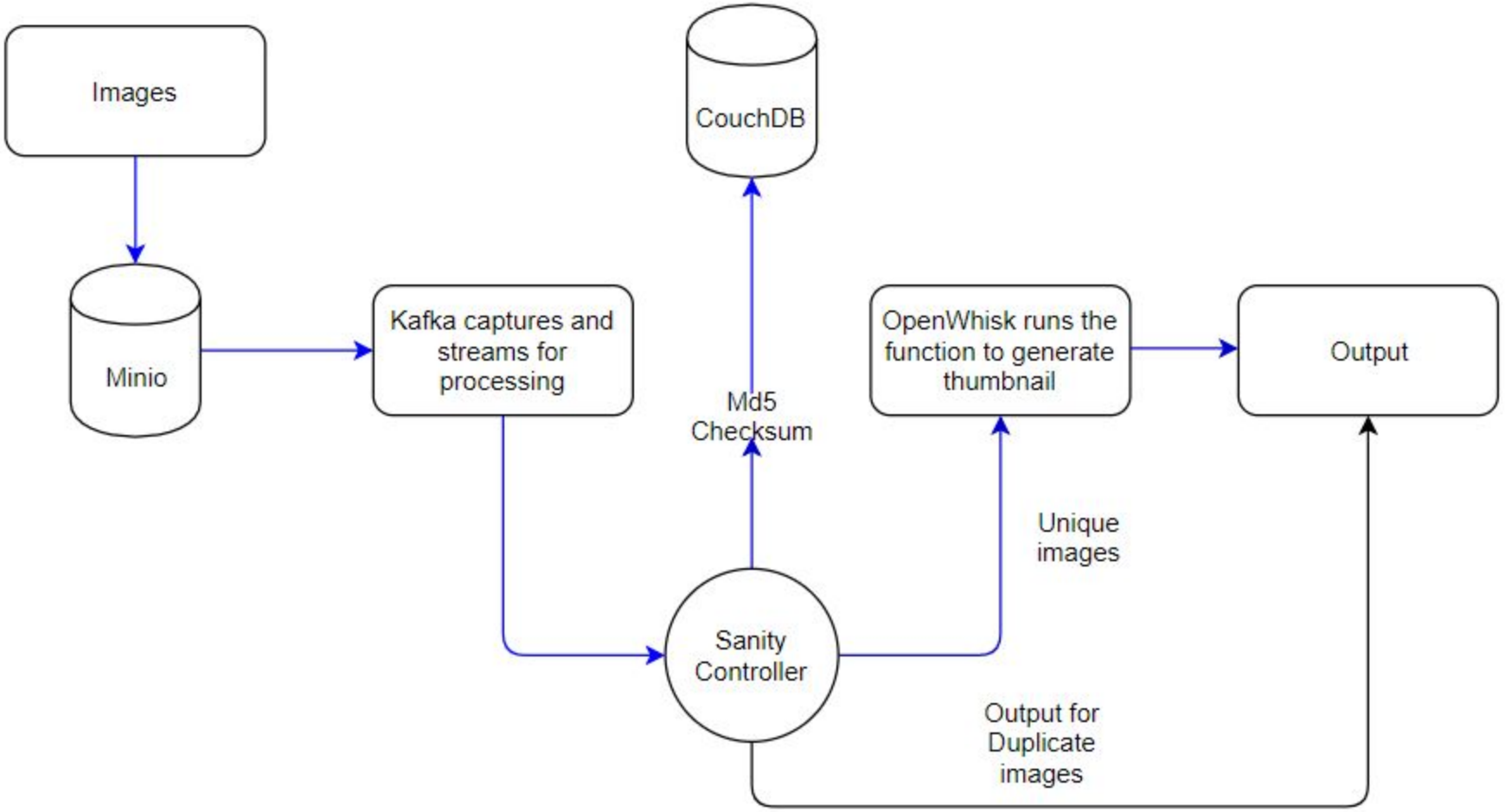


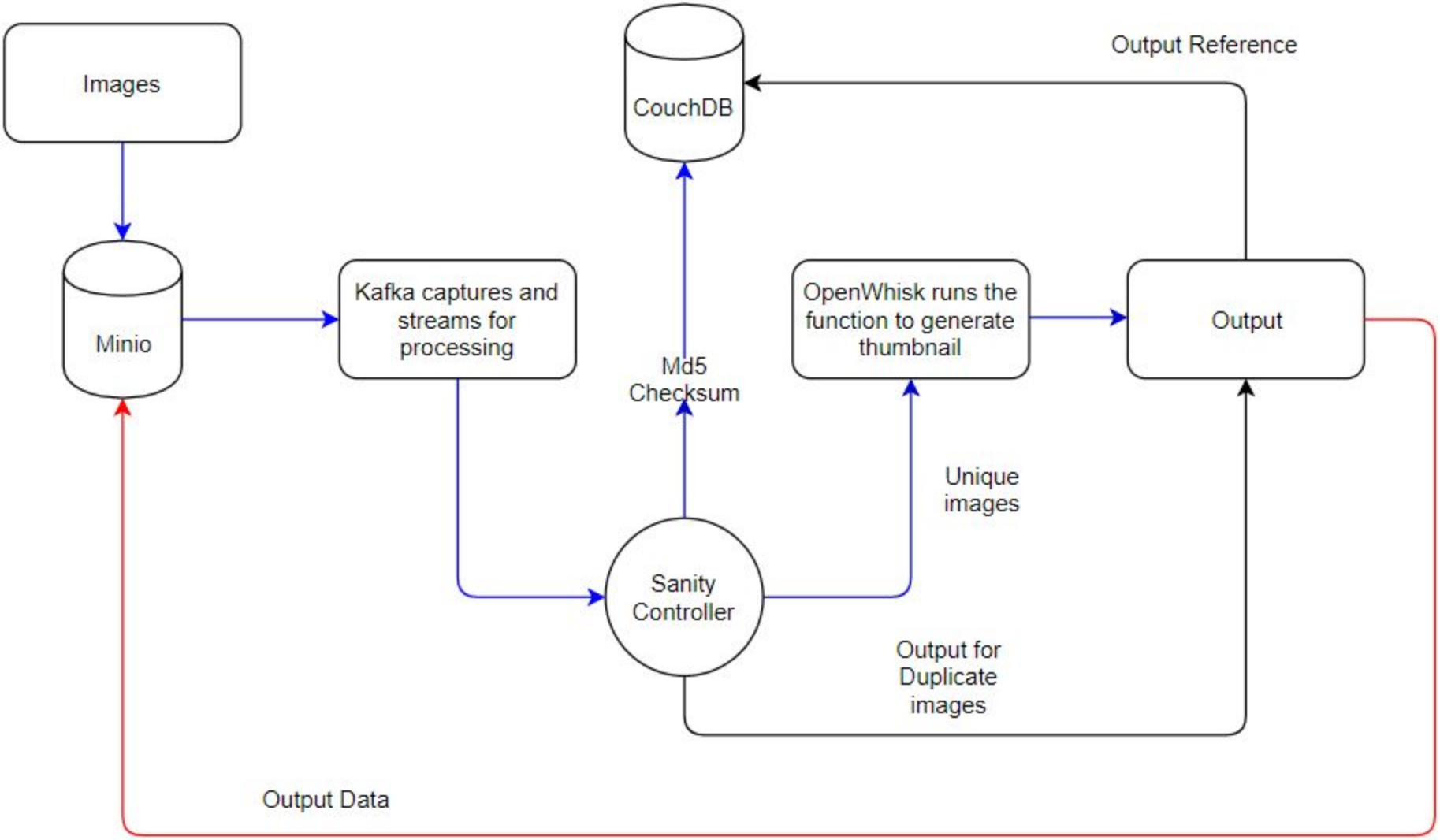

Images
*
Minio
*
Kafka captures and streams for processing
CouchDB
Md5 Checksum
Sanity Controller
Unique images
OpenWhisk runs the function to generate thumbnail
Output
Output for Duplicate images

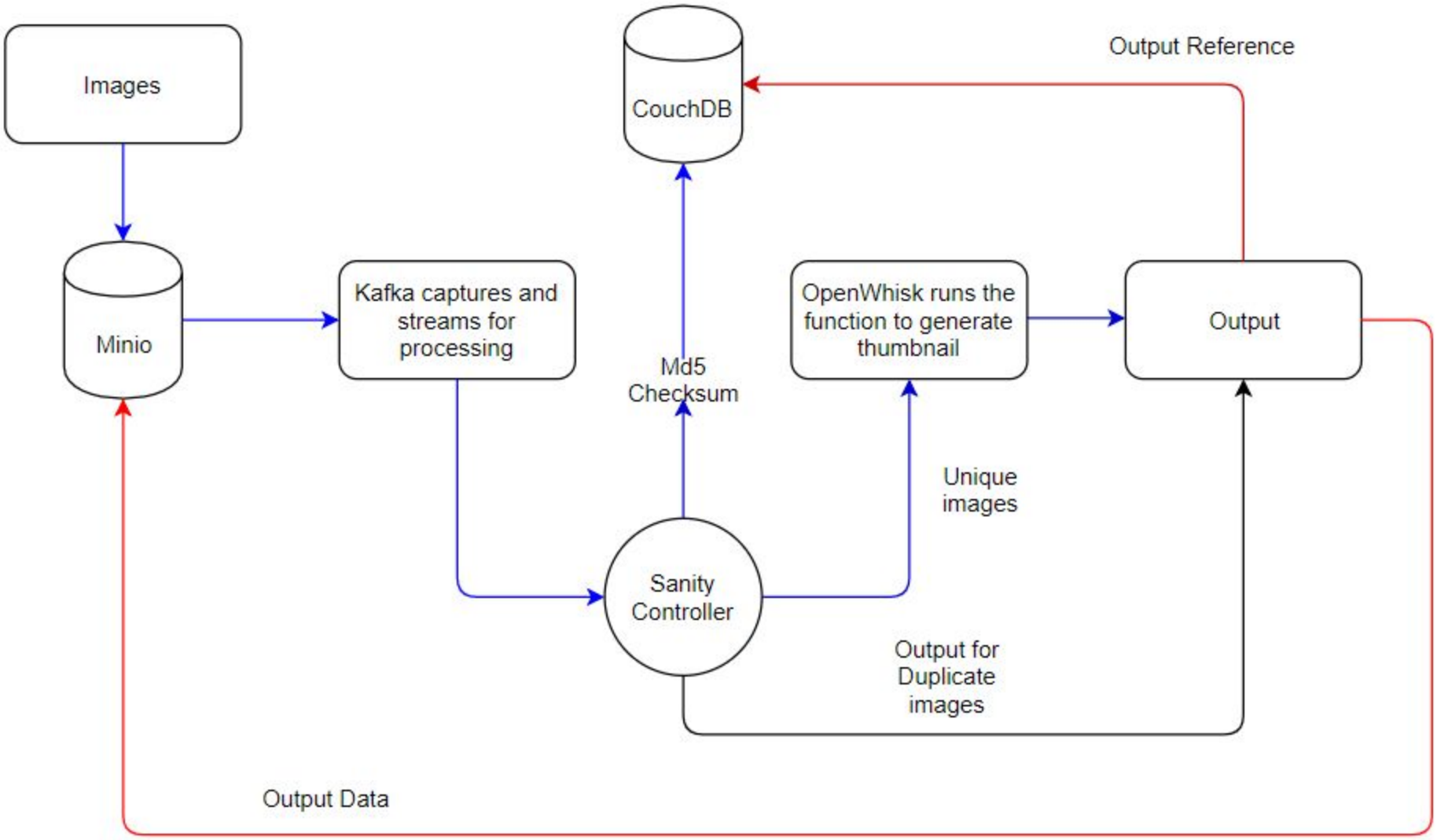

Images
*
Minio
*
Kafka captures and streams for processing
*
CouchDB
Md5 Checksum
Sanity Controller
Unique images
OpenWhisk runs the function to generate thumbnail
Output
Output for Duplicate images

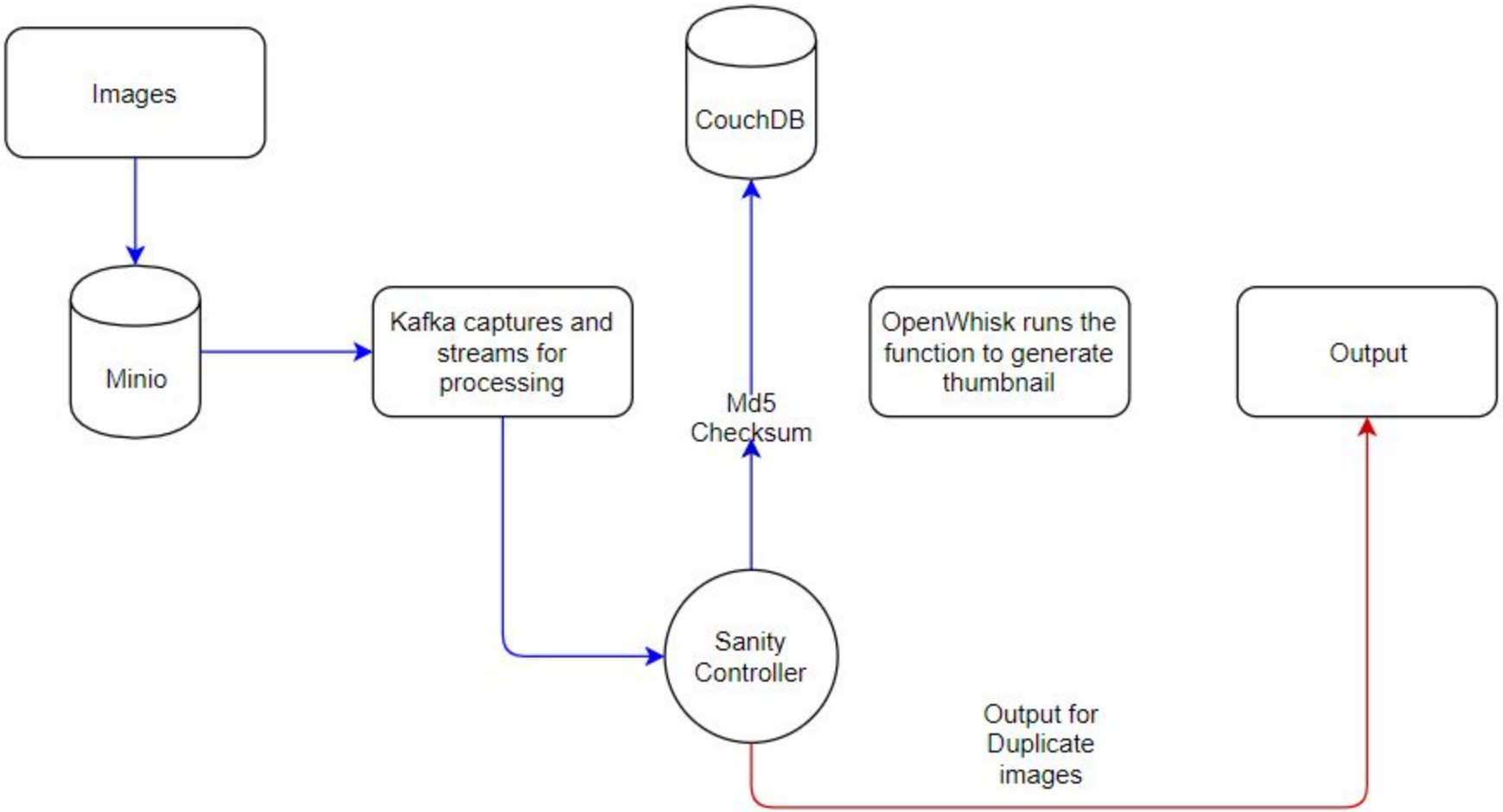

Images
*
Minio
*
Kafka captures and streams for processing
*
CouchDB
Md5 Checksum
Sanity Controller
Unique images
OpenWhisk runs the function to generate thumbnail
Output
Output for Duplicate images


Images
Minio
Kafka captures and streams for processing
CouchDB
Md5 Checksum
Sanity Controller
Unique images
OpenWhisk runs the function to generate thumbnail
Output
Output for Duplicate images




Images
Minio
Kafka captures and streams for processing
CouchDB
Output Reference
Md5 Checksum
Sanity Controller
Unique images
OpenWhisk runs the function to generate thumbnail
Output
Output for Duplicate images
Output Data




Images
Minio
Kafka captures and streams for processing
CouchDB
Md5 Checksum
Sanity Controller
OpenWhisk runs the function to generate thumbnail
Output
Output for Duplicate images

DEMO

How is deduplication achieved via Sanity Framework

- Upload a file to a Minio bucket (ex: inbucket, a.jpg)

  > mc cp a.jpg myminio/inbucket
- Create a function (ex: myfunction) with wsk client

  > wsk -i action create myfunction.py myfunction
- Invoke the function with the uploaded file

  > wsk -i action invoke myfunction inbucket/a.jpg outbucket
- Result is automatically stored in Minio bucket(ex: outbucket)
- For duplicate images, we get the reference from Sanity and for unique images, openwhisk executes action and stores the thumbnail, hence de duplication of functions is achieved.

Couch DB


```
{
  "_id": "dae770ad388c898fa85dc140a0014a24",
  "_rev": "5-2b4a80a884faf4eab778ab5902f5e3ae",
  "function1hash": {
    input1forfunction1hash: "outputbucketfor_input1forfunction1hash/outputfilenamefor_input1forfunction1hash",
    input2forfunction1hash: "outputbucketfor_input2forfunction1hash/outputfilenamefor_input2forfunction1hash"
  },
  "function2hash": {
   input1forfunction2hash: "outputbucketfor_input1forfunction2hash/outputfilenamefor_input1forfunction2hash",
   input2forfunction2hash: "outputbucketfor_input2forfunction2hash/outputfilenamefor_input2forfunction2hash"
  }
}
```


input_checksum

output_bucket/output_file

Deduplication In Progress..

Same file, different name!

BOSTON UNIVERSITY

After Running: Only one output for both files!

only output for test.jpg and duplication.jpg

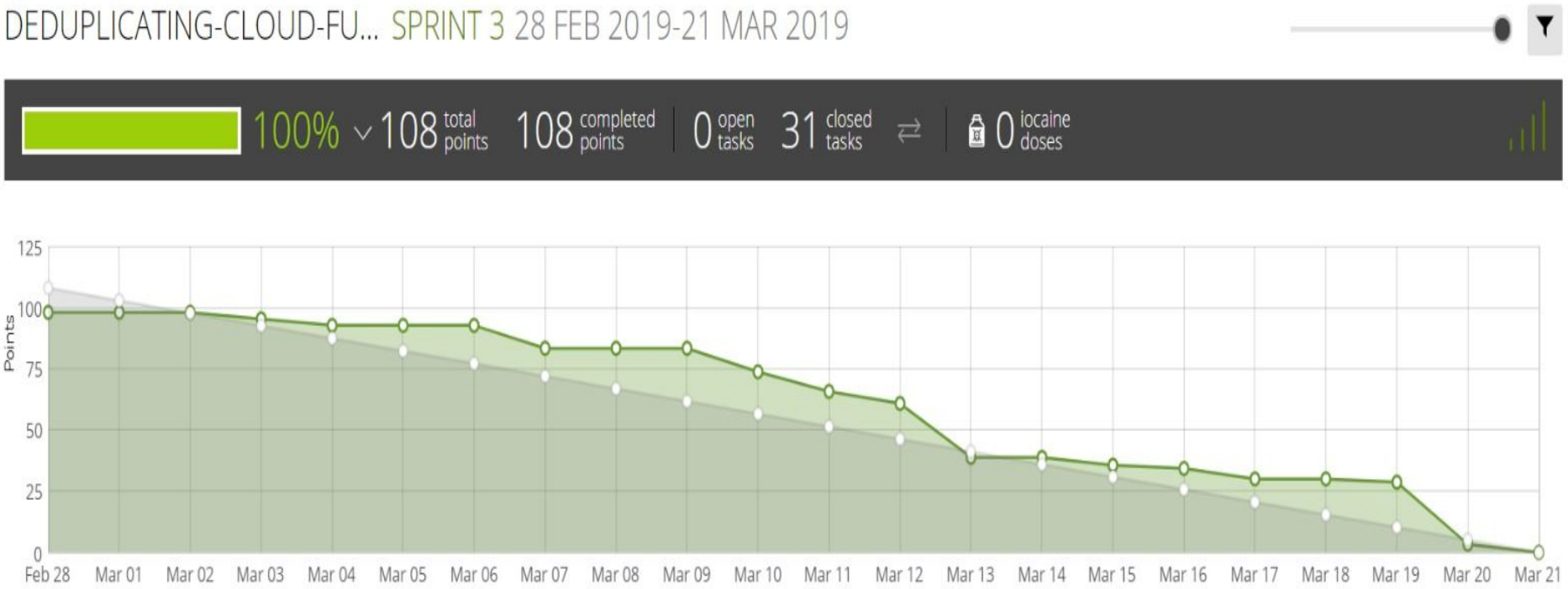
Challenges in Current Sprint

- Configuring docker components in cloud and executing functions in them
- How can we invoke custom docker scripts in openwhisk?
- Creating network connection between two servers in IBM cloud
- Simulating the entire pipeline with all the components in place in cloud

Next Steps (Sprint - 4)

- Implementing the Command Line Interface
- Benchmarking performance savings for entire process
- Brainstorm with our mentor to improve our current framework

Burndown Chart



THANK YOU