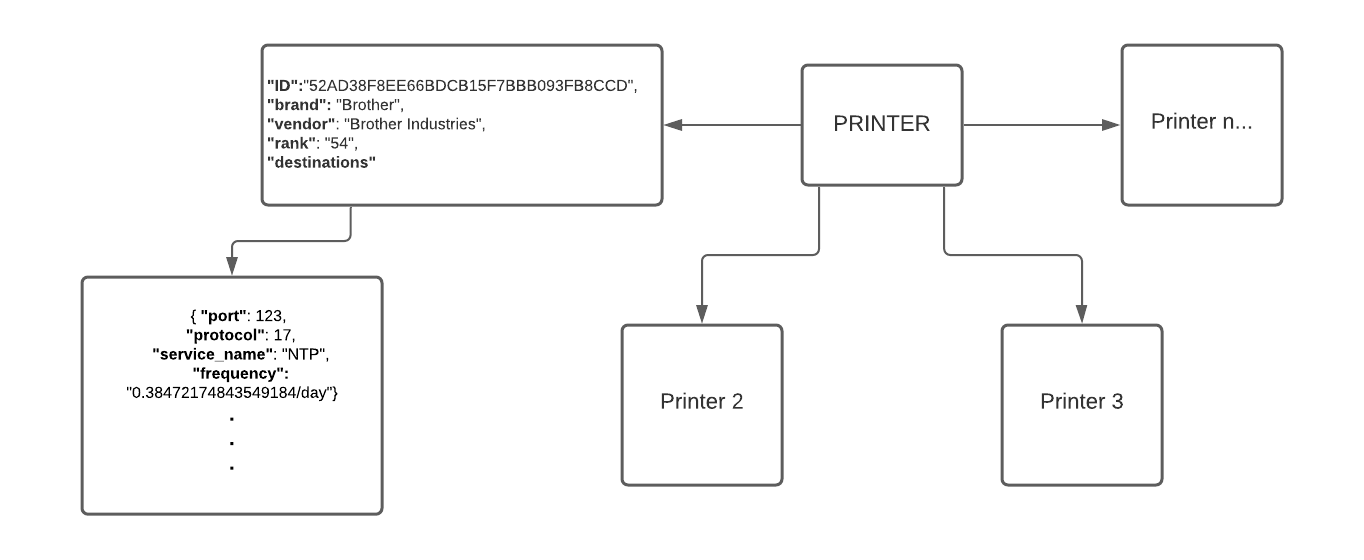
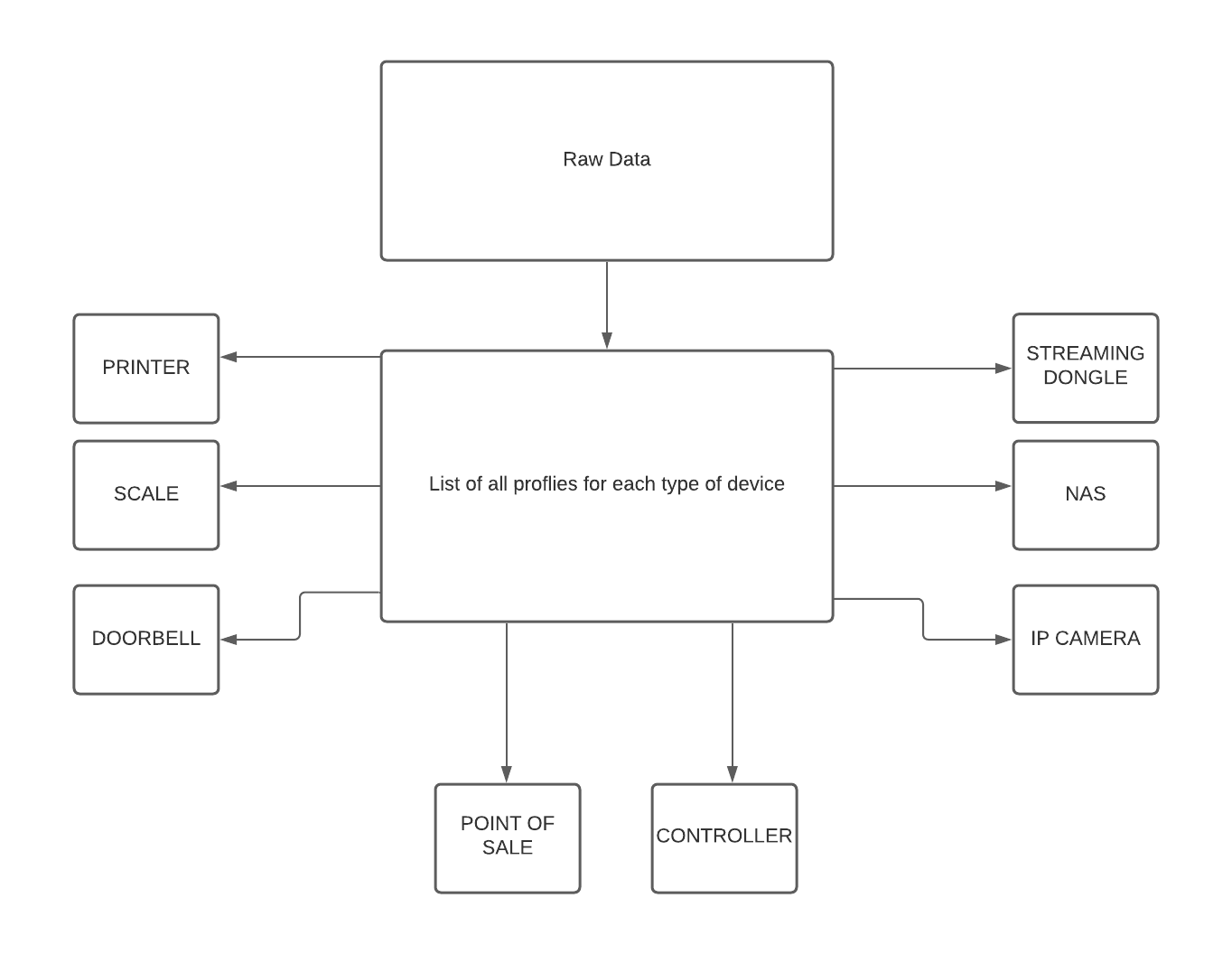
Work process:

* Understanding the components of the statistics for each IOT device.
* Researching and learning the concepts of networking that were included in the data for example the protocols and the layers in which the protocols belong…
* Deciding what in the data is relevant for building a profile:
  + Define the profile based on: Port, Protocol, Service name and the frequency per day. I've defined the frequency to be the number of connections divided by the length of time between first seen until last seen.
  + Filtering out IPs, host\_name. I've decided to filter these properties out because the destination host is not under my control and perhaps in the future the IPs may change (because of DHCP…) and thus the profile won't be relevant. For a similar reason host\_name can be changed at every moment.
* Arranging the data without duplicates.
* Writing the data to a JSON file.

Challenges:

* Understanding the hierarchy of the data structure.
* Handling duplicates.
* Runtime optimizations – In the code at some points I've used a list in order to search duplicates of python objects (dictionaries). Apparently, python dictionaries aren't hashable object and so cannot be used as set items.   
  I do have a solution (which I didn't implement because of lack of time). Creating a class that wraps the python dictionary and has a hash method. And then instead of using python dictionary I'll use my wrapper class in a set etc... which will definitely improve runtime of searching.
* Normalizing the frequency
* Ignoring time overlaps – Because of lack of time.

Ideas for other type differentiator or rank improvements:

The frequency property gives an indication of how important a communication stream (defined by port, service\_name, protocol) is to a device. The higher the frequency the more likely the communication stream is going to be used by the device in the future. For this reason, the frequency can help us improve the rank property. Perhaps, we can define a threshold that will help us filter out frequencies that aren't helpful for profiling a device. The frequencies that will remain will be used to calculate a new rank score. Perhaps it can be calculated by giving a score to each communication stream and with these scores calculate a weighted average that will be the rank.