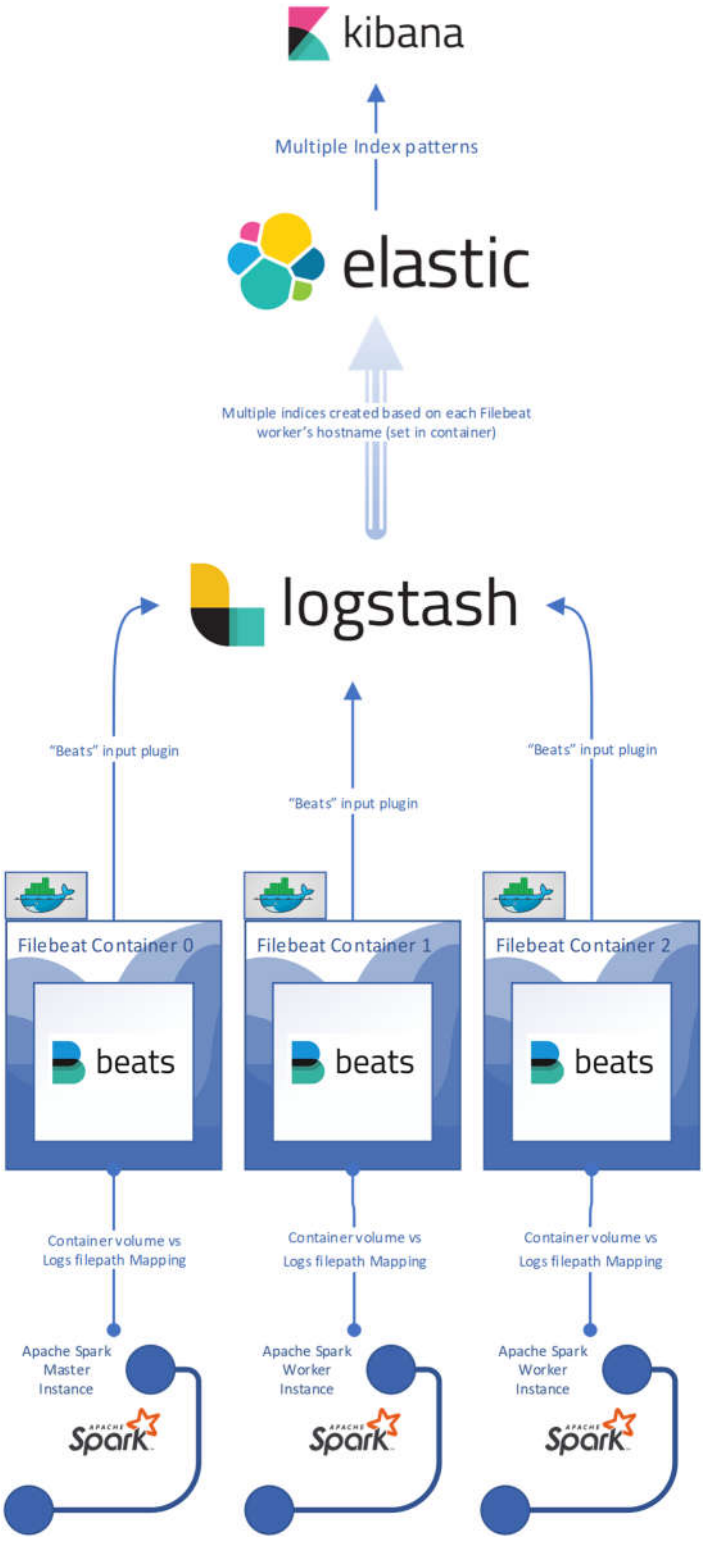
Architecture



Cluster setup details

- docker-compose.yml: Defines which containers will be created based on which image. Other settings are:
    - Environment variables to pass from host machine to containerized image (**environment_variables**)
    - Expose and map ports from container to host machine (**ports**)
    - Order of precedency during bootstrap (docker-compose up -d) achieved by **depends_on** keyword
    - Hostname of the containerized instance – similar to a VM (**hostname)**
    - Map host filepaths to container paths via **volumes** keyword
    - Define and assign common network settings for all containers -**network (s)**
- filebeat.yml: Defines the general settings that the Filebeat component needs in order to know which filepaths to monitor and where to send the collected information.
    - To add new inputs (log paths to monitor), edit filebeat.inputs section
    - To be able to create indices with different names later on, the **name** setting is set to the hostname (at the filebeat container level)
    - To send the collected information an **output.logstash** is defined, the default is to go directly to ElasticSearch, but we are using Logstash to enable future transformations in the logged data.
- kibana.yml: Defines general settings for the web application. Perhaps the most relevant one is the reference to a valid ElasticSearch cluster, usually if this elastic node is down, Kibana will not be able to load as well. Needs to be same version as ElasticSearch cluster being plugged to it.
- elasticsearch.yml: Defines Elastic general settings. Some relevant settings are: storage type (currently niofs to avoid mmapfs virtual memory allocation issues in Mac OS) and **network.host** that is usually set to the same hostname in which the Elastic node is running.
- jvm.options: Defines parameters relevant to the JVM (Java Virtual Machine) & GC (Garbage Collector). Usually memory-related. Relevant ones are: **-Xms512m & -Xmx512m** which represent the initial and maximum size of total space dedicated to the heap.
- Logstash pipelines: Logstash allows to specify many .conf files which then at bootstrap time are all concatenated into one single file by Logstash. Having separated config files allows to easily keep data pipelines and business use-cases in order, which the certainty that changes to one will not affect the other pipelines. Basic structure of each of them is to have a **input** section, **filter** section for transformation (invoking methods from the **input** plugin or writing **ruby** code) and **output** section . Input and output sections are invocations of an input and output Logstash plugin respectively. Currently, **beats** is the chosen input plugin (which tells Logstash to listen at port 5044 for beats data) in the three sample pipelines and **elasticsearch** is the output plugin for all of them. This is where the name of the new index is decided, based on the following code:
  `index => "%{[beat][name]}-%{[beat][version]}-%{+YYYY.MM.dd}"`
  beat is a field structure filled by filebeat when sending data to Logstash, and **name** is the actual beat name set in filebeat.yml (which is defaulted to hostname of filebeat worker/container)