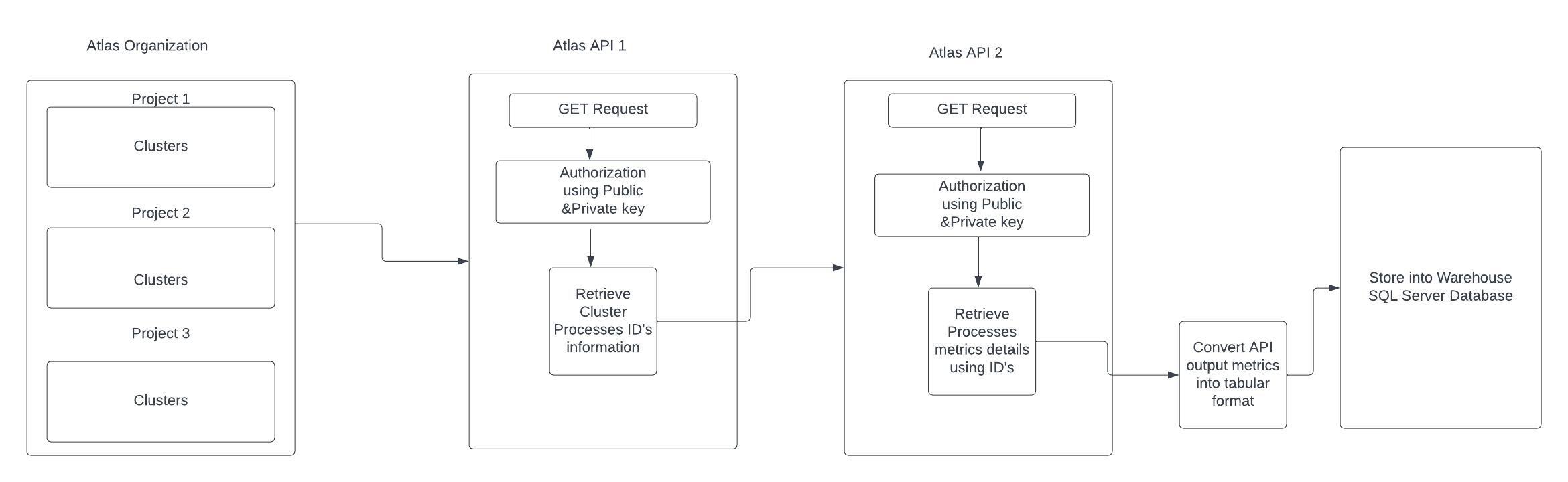
**Atlas Performance Model Architecture:**

**Approach 1:**

The Atlas Administration API follows the principles of the REST architectural style to expose a number of internal resources which enable programmatic access to Atlas's features.



**JSON entities:**

All entities are expressed in JSON.

**Key-based access:**

Each Atlas user or application needing to connect to Atlas must generate an API key before accessing the Atlas Administration API.

**Digest authentication:**

To ensure that your Atlas Administration API key is never sent over the network, Atlas Administration API requests are authenticated using HTTP Digest Authentication.

**HTTPS-Only:**

You can only access the Atlas Administration API via HTTPS, ensuring all data sent over the network is fully encrypted using TLS.

**User Access Control:**

Each Atlas user's Atlas Administration API capabilities match the permissions granted by their Atlas Atlas User Roles.

**API Network Access List:**

Atlas can secure access to its Atlas Administration API through an API access list. This list restricts access to the API to specific IP or CIDR addresses. Each Programmatic API Key has its own Atlas Administration API access list.

**Return All MongoDB Processes in One Project:**

<https://www.mongodb.com/docs/atlas/reference/api-resources-spec/v2/#tag/Monitoring-and-Logs/operation/listAtlasProcesses>

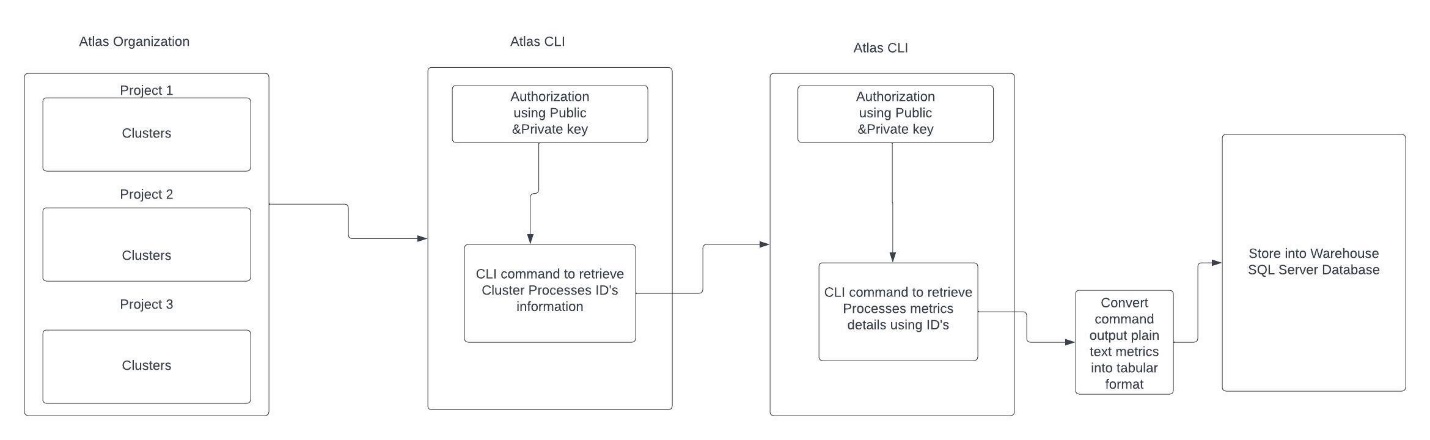
curl --user "publickey:privatekey" --digest --header "Accept: application/vnd.atlas.2023-02-01+json" “<https://cloud.mongodb.com/api/atlas/v2/groups/60f6ed508b8b1b6d8f916243/processes>”

**Return Measurements for One MongoDB Process:**

curl --user "publickey:privatekey" --digest --header "Accept: application/vnd.atlas.2023-02-01+json" "https://cloud.mongodb.com/api/atlas/v2/groups/60f6ed508b8b1b6d8f916243/processes/atlas-qtevhp-shard-00-02.hhn0w.mongodb.net:27017/measurements?granularity=PT1H&start=2024-01-15T00:00:00Z&end=2024-01-16T00:00:00Z"

**Approach 2:**

The Atlas CLI is a command line interface built specifically for MongoDB Atlas. We can use the Atlas CLI to easily manage Atlas from the terminal.

****

<https://www.mongodb.com/docs/atlas/cli/stable/atlas-cli-save-connection-settings/#save-connection-settings>

We can save connection settings as profiles. Profiles store the project IDs, organization IDs, and, optionally, API keys to use in future Atlas CLI sessions. You can also specify a profile instead of using the --projectId and --orgId flags with each command. The Atlas CLI stores your profiles in a configuration file called config.toml.

<https://www.mongodb.com/docs/atlas/cli/v1.0/command/atlas-processes-list/#atlas-processes-list>

# Return a JSON-formatted list of all MongoDB processes in the project with the ID 5e2211c17a3e5a48f5497de3:

atlas processes list --projectId 5e2211c17a3e5a48f5497de3 --output json

<https://www.mongodb.com/docs/atlas/cli/stable/command/atlas-metrics-processes/#atlas-metrics-processes>

atlas metrics processes testcluster-shard-00-02.hhn0w.mongodb.net:27017 --start 2024-01-19T00:00:00Z --end 2024-01-19T02:00:00Z --granularity PT5M