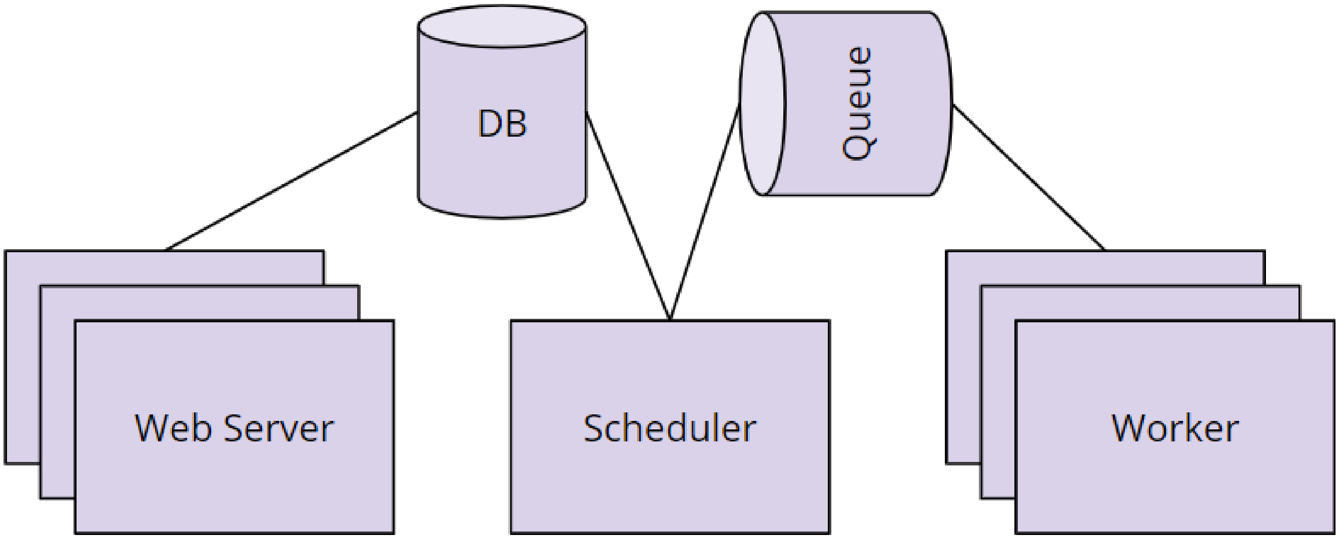
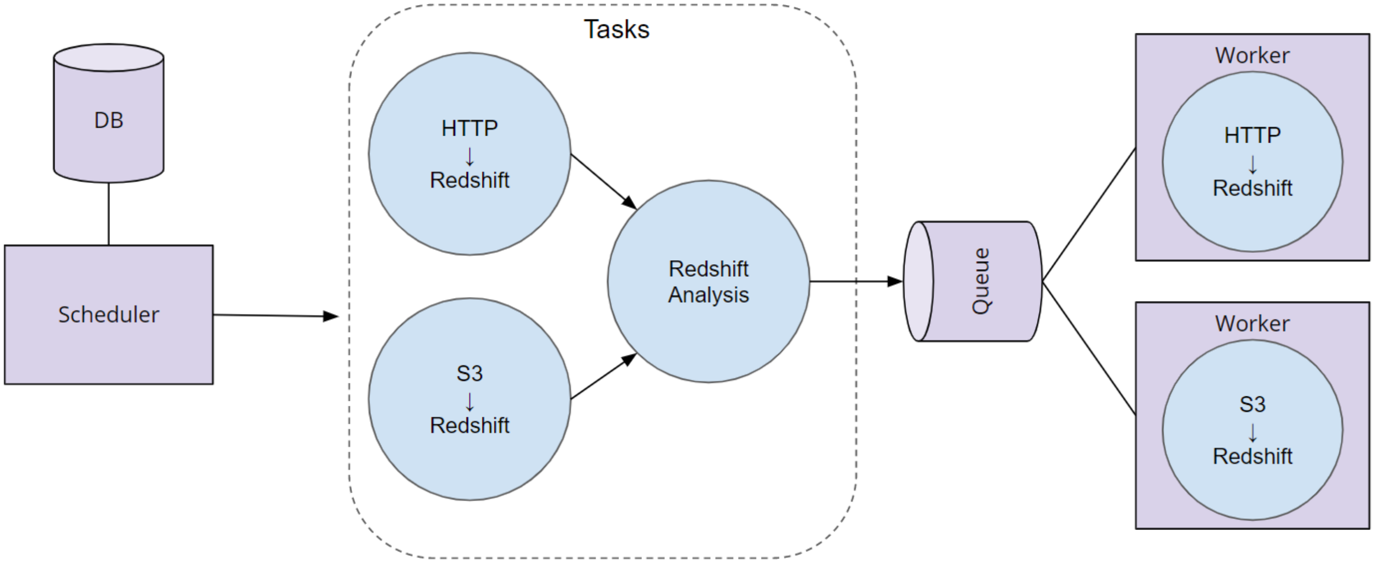
Components of Airflow



- **Scheduler** orchestrates the execution of jobs on a trigger or schedule. The Scheduler chooses how to prioritize the running and execution of tasks within the system. You can learn more about the Scheduler from the official **Apache Airflow documentation**.
- **Work Queue** is used by the scheduler in most Airflow installations to deliver tasks that need to be run to the **Workers**.
- **Worker** processes execute the operations defined in each DAG. In most Airflow installations, workers pull from the **work queue** when it is ready to process a task. When the worker completes the execution of the task, it will attempt to process more work from the **work queue** until there is no further work remaining. When work in the queue arrives, the worker will begin to process it.
- **Database** saves credentials, connections, history, and configuration. The database, often referred to as the *metadata database*, also stores the state of all tasks in the system. Airflow components interact with the database with the Python ORM, **SQLAlchemy**.
- **Web Interface** provides a control dashboard for users and maintainers. Throughout this course you will see how the web interface allows users to perform tasks such as stopping and starting DAGs, retrying failed tasks, configuring credentials, The web interface is built using the **Flask web-development microframework**.

How Airflow Works



Order of Operations For an Airflow DAG

- The Airflow Scheduler starts DAGs based on time or external triggers.
- Once a DAG is started, the Scheduler looks at the steps within the DAG and determines which steps can run by looking at their dependencies.
- The Scheduler places runnable steps in the queue.
- Workers pick up those tasks and run them.
- Once the worker has finished running the step, the final status of the task is recorded and additional tasks are placed by the scheduler until all tasks are complete.
- Once all tasks have been completed, the DAG is complete.