

# Stopping And Starting Stand-Alone AtScale

This document describes how to stop and start Stand-Alone AtScale and AtScale services.

## Before You Begin

- ▶ These instructions are for CentOS/RedHat operating systems. You may have to modify for your supported OS.
- ▶ Unless otherwise specified, run commands as the atscale user.

## About This Task

You can list AtScale services, review their logs, and stop and start individual AtScale services as well as the AtScale application.

AtScale bundles Supervisor <http://supervisord.org> to use in managing all applications and services used to run the product. You work with AtScale's Supervisor via the `atscale_service_control` script.

## Procedure

- ▶ List AtScale services

To list all the services that AtScale is managing, including their status:

1. Run `atscale_service_control status`

```
$ /opt/atscale/bin/atscale_service_control status
account                RUNNING   pid 6011, uptime 0:01:09
coordinator            RUNNING   pid 5270, uptime 0:01:47
engine                 RUNNING   pid 7573, uptime 0:00:24
haproxy                RUNNING   pid 5371, uptime 0:01:43
modeler                RUNNING   pid 5907, uptime 0:01:15
patroni                RUNNING   pid 5546, uptime 0:01:35
servicecontrol         RUNNING   pid 6187, uptime 0:00:57
sidecarserver          RUNNING   pid 6116, uptime 0:01:01
```

- ▶ Review logs

Review the AtScale service logs.

The logs for each of the AtScale-managed services are in the `/opt/atscale/log/` directory with the naming convention `<SERVICE_NAME>.stdout`. Individual applications and services may have additional logging, which can be in subdirectories of `/opt/atscale/log/`.

- ▶ Stop individual services

To stop a single AtScale-managed service:

1. Determine the name of the service via the `atscale_service_control status` command.
2. Issue the `stop` command.

```
$ /opt/atscale/bin/atscale_service_control stop engine engine: stopped
```

You can run this command repeatedly without impacting the services that are already stopped.

#### ▲ Stop AtScale "apps"

To stop all the AtScale apps, which are AtScale-developed services, run the `atscale_stop_apps` script.

```
$ /opt/atscale/bin/atscale_stop_apps
Running this will stop all AtScale apps on this machine.
Supporting services, such as AtScale's Postgres, will remain running.
Continue? (y/n): y
Stopping AtScale applications...
account: stopped engine: ERROR (not running)
modeler: stopped
servicecontrol: stopped
sidecarserver: stopped
AtScale applications have been stopped.
```

You can run this command repeatedly without impacting the services that are already stopped.

#### ▲ Fully Stop AtScale

To stop AtScale entirely, shutting down all apps and services as well as Supervisor, run the `atscale_stop` script.

```
$ /opt/atscale/bin/atscale_stop
Running this will fully stop all AtScale apps and services on this machine.
Continue? (y/n): y
Stopping all AtScale apps and services...
Stopping AtScale applications...
account: ERROR (not running)
engine: ERROR (not running)
modeler: ERROR (not running)
servicecontrol: ERROR (not running)
sidecarserver: ERROR (not running)
AtScale applications have been stopped.
patroni: stopped
haproxy: stopped
coordinator: stopped
Shut down
AtScale has been stopped.
```

You can run this command repeatedly without impacting the services that are already stopped.

#### ▲ Start AtScale from a full stop

To start AtScale from a fully stopped state:

1. As a precautionary measure before starting up AtScale, use the `ps` command to check if any AtScale processes still running.

```
$ ps -aef | grep /opt/atscale
atscaler 3848 3789 0 20:00 ?      00:00:07 python3
/opt/atscale/versions/7.4.0.1233/pkg/python3/bin/patroni
/opt/atscale/versions/7.4.0.1233/conf/database/patroni.yml
```

2. Kill them if needed prior to restarting.

```
$ kill -9 2149
$ ps -aef | grep /opt/atscale
atscaler 597 517 0 02:50 pts/0 00:00:00 grep /opt/atscale
```

3. Run `atscale_service_control status` to determine that AtScale is completely stopped.
4. Run the `atscale_start` script to start AtScale.

```
$ /opt/atscale/bin/atscale_start
```

You can run this command repeatedly without impacting the services that are already stopped.

1. Check the status after starting up:

```
$ /opt/atscale/bin/atscale_service_control status
account          RUNNING  pid 13009, uptime 0:00:11
coordinator      RUNNING  pid 13024, uptime 0:00:11
engine           RUNNING  pid 13013, uptime 0:00:11
haproxy          RUNNING  pid 13019, uptime 0:00:11
modeler          RUNNING  pid 13007, uptime 0:00:11
patroni          RUNNING  pid 13003, uptime 0:00:11
servicecontrol   RUNNING  pid 13015, uptime 0:00:11
sidecarserver    RUNNING  pid 13014, uptime 0:00:11
```

#### ▲ Start AtScale apps

To start all the AtScale apps (the AtScale-developed services), run the `atscale_start_apps` script.

```
$ /opt/atscale/bin/atscale_start_apps
Starting AtScale applications...
account: started
engine: started
modeler: started
servicecontrol: started
sidecarserver: started
AtScale applications have been started.
```

You can run this command repeatedly without impacting the services that are already started.

## ▲ Start individual services

To start a single AtScale-managed service:

1. Determine the name of the service via the `atscale_service_control status` command.
2. Run the `start` command.

```
$ /opt/atscale/bin/atscale_service_control start engine  
engine: started
```

These commands can run repeatedly without impacting the services that are already started.