

Upgrading

Generated on September 26, 2026

An upgrade replaces the application and nothing else: your configuration, your data, the DataOrchestrator's identity and its licence all stay where they are.

[!IMPORTANT] Read the release notes before you start. When a release stops trusting the key your licence was signed with, an online DataOrchestrator has to reach the portal as it starts, and an air-gapped one needs its offline key replaced within two hours of starting, or it stops until the key is registered — see *Your licence after the upgrade* below.

Before you start: back up

Back up these four directories before every upgrade, without exception:

- `entities/`
- `logs/`
- `loggers/`
- `instance/` — the DataOrchestrator's identity and its licence, which exist nowhere else

An in-place upgrade replaces the application binaries only; it must never touch these directories. Taking the backup regardless costs you a few minutes and protects against the one bad release that breaks this assumption.

In-place upgrade

1. Download the new installer for your platform from the [download page](#).
2. Stop the running service (`systemctl stop data-orchestrator` , or the equivalent for your platform).
3. Run the new installer over the existing install — it replaces the application files and leaves `entities/` , `logs/` , `loggers/` and `instance/` untouched.
4. Start the service again and confirm the version shown in the UI matches what you installed.
5. Open the licence pane and confirm the licence is in force: **Licensed**, with no warning. If it is not, see *Your licence after the upgrade* below.
6. Confirm your existing dashboards, modules and peers all still load and report data as before.

Your licence after the upgrade

An upgrade keeps the licence: the DataOrchestrator starts on the licence it held, and nothing has to be registered again.

The exception is a release that stops trusting the key your licence was signed with. The licence pane then says *The licence on this machine was signed with a key this version of DataOrchestrator no longer trusts*. The engine starts all the same, but it runs only the two hours of that start until it holds a licence signed with the current key:

- **Online** — keep the portal reachable while the DataOrchestrer starts. It collects a new licence by itself, usually within seconds, and restarts once, shortly after starting, when its new licence arrives: it loads its configuration again, so everything the licence enables applies at once. If it collects none, restore its connection and select **Check for licence**: an engine that stopped in the meantime starts again by itself once the licence arrives.

- **Air-gapped** — replace the offline key. Plan the maintenance window around it: the engine runs for two hours from the start after the upgrade, and if the new key is not registered by then, it stops until it is.

1. In the licence pane of the upgraded DataOrchestrer, select **Copy DO Id**. Copy it after the upgrade: a DO Identifier copied before it gets a key the new version cannot use.
2. On the licence in the portal, select **Replace offline key**, paste the DO Identifier and copy the new key.
3. In the licence pane, select **Register offline** and paste the key. An engine that stopped starts again by itself; one still in its two hours restarts once, loading its configuration again.

The button reads **Register offline**, not **Register new key**: a licence signed with a key that is no longer trusted is no licence, so the DataOrchestrer holds none.

If `instance/` was lost, restore it from the backup. Without a backup, an online DataOrchestrer collects its licence again at its next check, and an air-gapped one needs its offline key registered again: **Copy offline key** on the licence, then **Register offline** in the licence pane. If the licence pane shows a different install ID than before, the identity was lost as well: restore `instance/`, `entities/` and `loggers/` from the backup before doing anything else.

When licensing stops an upgraded engine

The release that runs every start for two hours also changes when licensing stops an engine. Once upgraded, a DataOrchestrer:

- starts whatever its licence says, and runs for two hours from every start; after that, only a licence in force keeps it running — see [How licensing works](#);
- with an online licence, stops seven days after the portal last confirmed that licence: plan any work that cuts it off from the portal around those seven days;
- stops within about four hours of being cleared in the portal.

A DataOrchestrer not yet upgraded keeps the rules it was built with: it does not start without a licence, and an online one stops at the end of its lease plus its grace — up to about two weeks after it last reached the portal, or after it was cleared. Whichever version runs, a lease or offline key signed since the grace was shortened carries 7 days of grace after its expiry; an offline key cut before keeps its 14 days.

What must not happen

- Configuration should not need re-entering after an upgrade.
- No entity file should be silently rewritten by the new version on first load — if you see every entity file's modification time change right after an upgrade, that's a bug, not expected behaviour.

Rolling back

If something goes wrong, stop the service, reinstall the previous version's package, and restore `entities/`, `logs/` and `loggers/` from the backup you took before the upgrade if the new version wrote anything unexpected to them. A same-version reinstall never requires restoring the backup; only do so if you have a concrete reason to believe the new version altered your data.

A version from before a signing-key change does not trust licences signed with the new key. An online DataOrchestrator collects one it trusts at its next check, for as long as the portal still signs for that version; an air-gapped one needs `instance/` restored from the backup taken before the upgrade. Correct the machine's clock before you roll back: an older version refuses to start when the clock is behind the time the newer one trusted.

Next steps

- [Licence activation](#)
- [Uninstalling](#)