



TEIL I · GRUNDLAGEN

Der Ordivis Stack

Wie Ordivis Updater, Ordivis Service und die Fachanwendungen
zusammengehören

Produktversion 2026.09.18 · Handbuch-Revision 1.0
Stand 18.09.2026 · Plattform .NET 10 / Windows Server

OU-01 Stack

Der Ordvis Stack

Dieses Kapitel erklärt, wie eine Ordvis-Installation aufgebaut ist und welchen Platz **Ordvis Updater** darin einnimmt. Es richtet sich an **Administratoren**, die die Anlage betreiben.

Die Installationsschritte stehen in *Einrichten* (OU-10), der laufende Betrieb in *Aktualisieren* (OU-30), der Ernstfall in *Wenn ein Update schiefgeht* (OU-50).

INHALT

- 1 Was „Ordvis Stack“ bedeutet
- 2 Was Ordvis Updater tut
- 3 Wie eine Anwendung anschlussfähig wird
- 4 Die Verbindungen
- 5 Was Ordvis Updater nie anfasst
- 6 Signaturen

1 Was „Ordvis Stack“ bedeutet

Eine Ordvis-Installation besteht nicht aus einem Programm, sondern aus **Schichten**. Jede setzt auf der darunter auf – wie beim OSI- oder TCP/IP-Modell. Für diesen Aufbau verwenden wir den Begriff **Ordvis Stack**.

Der Ordvis Stack

Was unten liegt, trägt das darüber – gelesen wie OSI oder TCP/IP.

Außerhalb der Anlage

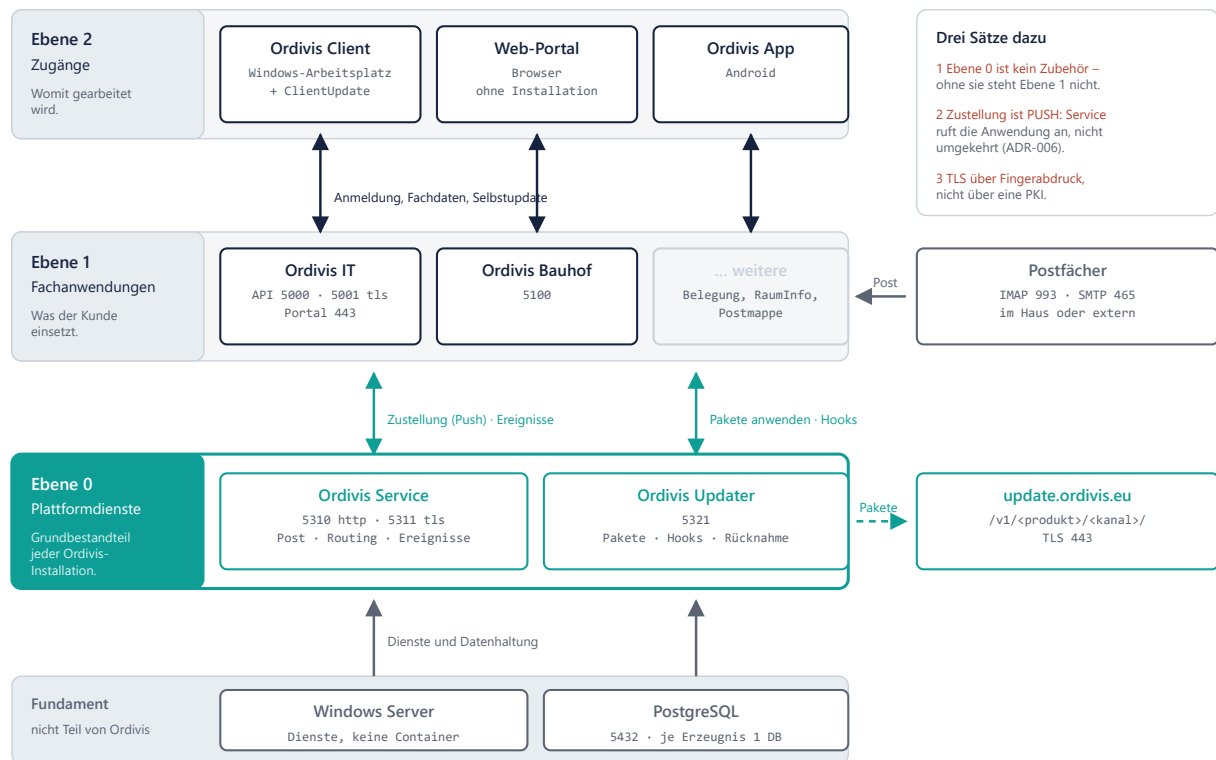


Abb. 1 – Der Ordvis Stack. Was unten liegt, trägt das darüber. Ordvis Updater bildet zusammen mit Ordvis Service die Ebene 0.

Ebene	Was dort liegt	Beispiel
2 – Zugänge	Womit gearbeitet wird	Ordvis Client, Web-Portal, App
1 – Fachanwendungen	Was der Kunde einsetzt	Ordvis IT, Ordvis Bauhof
0 – Plattformdienste	Der gemeinsame Unterbau	Ordvis Service, Ordvis Updater
<i>Fundament</i>	Nicht Teil von Ordvis	Windows Server, PostgreSQL

Ebene 0 ist kein Zubehör. Ordvis Updater ist ein **Grundbestandteil jeder Ordvis-Installation**. Ohne ihn aktualisiert sich jede Fachanwendung selbst – mit eigenem Sicherungsverfahren, eigener Rücknahme und eigenem Zeitpunkt. Bei zwei Anwendungen ist das unübersichtlich, bei fünf nicht mehr zu verantworten.

2 Was Ordvis Updater tut

Er ist der **Wartungsdienst** der Anlage:

Schritt	Was geschieht
Bestand aufnehmen	Welche Ordvis-Anwendungen liegen auf diesem Rechner, in welcher Fassung
Katalog abgleichen	Was der Hersteller anbietet
Planen	Was in welcher Reihenfolge zu tun ist – mit Abhängigkeiten
Freigeben	Ein Mensch bestätigt mit Namen; die Freigabe ist ein Nachweis
Anwenden	Sichern, Dienst anhalten, tauschen, migrieren, prüfen
Zurücknehmen	Wenn die Prüfung fehlschlägt – selbsttätig

⚠ Inventar und Katalog sind zwei verschiedene Dinge, und die Unterscheidung ist wichtig.

sagt	
Inventar	„Ich <i>sehe</i> diese Installation“ – die Bestandsaufnahme findet jede auf dem Rechner
Katalog	„Ich <i>kann</i> sie aktualisieren“ – dafür braucht es ein signiertes Paket im Feed

Eine Anwendung, die im Inventar steht, aber nicht im Katalog, kann der Updater **nicht** versorgen.

3 Wie eine Anwendung anschlussfähig wird

Ordvis Updater braucht keinen Code in der Fachanwendung. Sie liefert eine **Beschreibung** mit — `ordvis-hooks.json`, im selben Verzeichnis wie ihre Programmdatei:

```
{
  "productId": "bauhof",
  "phasen": {
    "pre": [ { "art": "Datenbanksicherung", "verbindung": "haupt" } ],
    "migrate": [ { "art": "Dateipruefung", "datei": "Ordvis.Bauhof.Api.dll" } ],
    "healthcheck": [ { "art": "Http", "adresse": "http://localhost:5100/gesundheit" } ],
    "rollback": [ { "art": "Datenbankwiederherstellung", "verbindung": "haupt" } ]
  }
}
```

Drei Regeln, an denen es sonst scheitert:

- ⚠ **healthcheck** ist Pflicht und muss fachlich prüfen – Datenbank erreichbar, Schema auf Stand. „Prozess läuft“ genügt nicht: Er ist die *einzige* Aussage, auf die sich der Updater bei „hat es geklappt?“ stützt.
- ⚠ Wer **migrate** zusagt, muss **rollback** zusagen. Eine Anwendung, die vorwärts migriert und nicht zurückkann, macht den Rückweg des **ganzen Laufs** unmöglich.
- ⚠ Die Sicherung gehört in **pre**. Dort läuft die Anwendung noch, und es ist nichts angefasst — schlägt sie fehl, endet der Lauf, bevor er begonnen hat.

4 Die Verbindungen

Von	Nach	Port / Adresse
Ordvis Updater	<code>update.ordvis.eu</code>	TLS 443, ausgehend
Administrator	Verwaltung Updater	5321
Arbeitsplatz-Client	Updater (Ringe, optional)	5321
Ordvis Updater	Dienste der Anwendungen	örtlich, kein Netz

★ Nur eine Verbindung verlässt die Anlage, und sie geht hinaus. Von außen muss nichts hereinkommen.

5 Was Ordvis Updater nie anfasst

Muster	Warum
<code>appsettings*.json</code>	Token, Anbindung, Verbindungszeichenfolgen
<code>*.key</code>	Schlüssel der Anlage
<code>data/**</code>	Nutzdaten
<code>server-url.txt</code>	Serveradresse, je Maschine
<code>*.db</code>	örtliche Datenbanken

⚠ Diese Regel gilt für das Anwenden UND die Nachprüfung – als eine Regel, nicht als zwei Listen. Als getrennte Listen führte das dazu, dass die Prüfung eine geschützte Datei für „fehlt“ hielt und eine Rücknahme auslöste. Bei jedem Lauf, endlos.

6 Signaturen

Pakete sind **ES256-signiert**, und die Prüfung ist nicht abschaltbar. Jedes Produkt hat ein **eigenes** Schlüsselpaar; der Katalog nennt bei jedem Erzeugnis seine `keyId`.

★ Ein Schlüssel für alles wäre ein **Generalschlüssel**. Wer den Schlüssel eines Produkts erbeutet, kann damit gefälschte Pakete *dieses* Produkts unterschreiben – aber keine anderen.

⚠ Der **private Schlüssel erreicht keine Anlage**. Er bleibt beim Hersteller. Auf der Anlage liegt ausschließlich der öffentliche Teil, unter `C:\ProgramData\Ordvis\Updater\schluessel\`.