



TEIL I · GRUNDLAGEN

Der Ordivis Stack

Wie Ordivis Service, Ordivis Updater und die Fachanwendungen
zusammengehören

Produktversion 2026.09.07 · Handbuch-Revision 1.0
Stand 07.09.2026 · Plattform .NET 10 / Windows Server

OS-01 Stack

Der Ordvis Stack

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

Die Installationsschritte stehen in *Einrichten* (OS-10), der laufende Betrieb in *Betrieb und Störungssuche* (OS-40).

INHALT

- 1 Was „Ordvis Stack“ bedeutet
- 2 Warum es diese Ebene gibt
- 3 Ordvis Service in einem Absatz
- 4 Ordvis Updater in einem Absatz
- 5 Die Verbindungen – und ihre Richtung
- 6 Was ausfällt, wenn Ebene 0 ausfällt

1 Was „Ordvis Stack“ bedeutet

Eine Ordvis-Installation besteht nicht aus einem Programm, sondern aus **Schichten**. Jede Schicht setzt auf der darunter auf und benutzt sie – 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

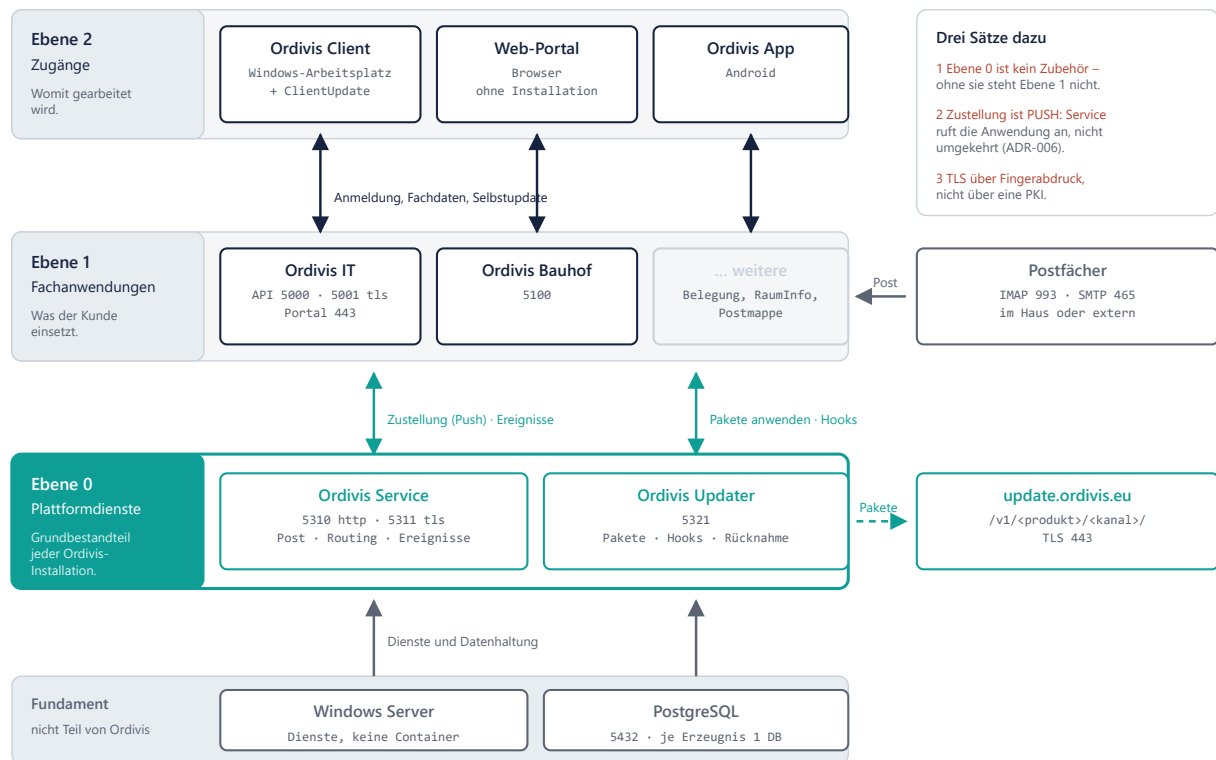


Abb. 1 – Der Ordvis Stack. Was unten liegt, trägt das darüber. Alle Wege verlaufen zwischen benachbarten Ebenen; nach außen führen nur zwei Verbindungen.

Abbildung 1 zeigt die drei Ebenen und das Fundament, auf dem sie stehen.

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
Fundament	Nicht Teil von Ordvis	Windows Server, PostgreSQL

Ebene 0 ist kein Zubehör. Ordvis Service und Ordvis Updater sind **Grundbestandteile jeder Ordvis-Installation**. Eine Fachanwendung ohne sie läuft zwar, aber sie holt ihre Post selbst ab, versendet selbst und aktualisiert sich selbst – und jede weitere Anwendung täte dasselbe noch einmal. Genau das soll der Stack beenden.

2 Warum es diese Ebene gibt

Der Grund ist nicht Ordnungsliebe, sondern eine Beobachtung aus dem Betrieb: **Was jede Anwendung für sich löst, löst sie anders.**

Ein Beispiel aus der Praxis. Zwei Anwendungen holen dasselbe Postfach ab. Beide funktionieren für sich, und trotzdem geht Post verloren: Wer zuerst kommt, nimmt die Mail – der andere sieht sie nie. Der Fehler ist nicht in einer der beiden Anwendungen zu finden, weil er in keiner steckt.

Ebene 0 nimmt den Anwendungen deshalb drei Aufgaben ab:

Aufgabe	Wer sie übernimmt	Was die Anwendung davon merkt
Post abholen, filtern, zuordnen	Ordvis Service	Sie bekommt fertige Zustellungen
Post versenden	Ordvis Service	Sie meldet Ereignisse, keine Mailtexte
Aktualisieren, sichern, zurücknehmen	Ordvis Updater	Sie liefert eine Beschreibung mit

3 Ordvis Service in einem Absatz

Ordvis Service ist die **Poststelle** der Anlage. Er holt die Postfächer ab – einmal je Postfach, nicht einmal je Anwendung –, entscheidet nach Regeln, welche Anwendung zuständig ist, und stellt ihr die Meldung zu. Umgekehrt nimmt er von den Anwendungen **Ereignisse** entgegen („Vorgang angelegt“, „Vorgang geschlossen“) und macht daraus die Post an den Melder.

⚠ **Ein Umschlag ist kein Vorgang.** Ordvis Service führt keinen Vorgangslebenszyklus – er weiß nicht, was ein Ticket ist. Er weiß, dass eine Meldung eingegangen ist, an wen sie ging und was die Anwendung daraus gemacht hat. Wer den Vorgang sucht, sucht ihn in der Anwendung.

4 Ordvis Updater in einem Absatz

Ordvis Updater ist der **Wartungsdienst** der Anlage. Er kennt jede installierte Ordvis-Anwendung, holt Pakete vom Hersteller, hält die Dienste an, tauscht die Dateien, prüft das Ergebnis – und nimmt zurück, wenn die Prüfung fehlschlägt.

Damit er eine Anwendung bedienen kann, liefert diese eine Beschreibung mit (`ordvis-hooks.json`): darin steht, was vor dem Update zu sichern ist, wie migriert wird und woran man erkennt, dass es geklappt hat.

Ausführlich steht das im **Administratorhandbuch von Ordvis Updater**.

5 Die Verbindungen – und ihre Richtung

Für die Firewall ist nicht wichtig, *dass* zwei Bausteine reden, sondern **wer anruft**.

Von	Nach	Port / Adresse
Ordvis Service	Postfächer	IMAP 993, SMTP 465

Von	Nach	Port / Adresse
Ordvis Service	Fachanwendung	deren HTTPS-Port, z. B. 5001
Fachanwendung	Ordvis Service	5310 (http) / 5311 (https)
Ordvis Updater	<code>update.ordvis.eu</code>	TLS 443, ausgehend
Administrator	Verwaltung Service	5310 / 5311
Administrator	Verwaltung Updater	5321
alle	PostgreSQL	5432

⚠ **Die Zustellung ist ein Push, kein Pull.** Ordvis Service ruft die Anwendung an, nicht umgekehrt. Wer die Firewall nach dem Bauchgefühl öffnet, öffnet die falsche Richtung – und sucht den Fehler dann in der Anwendung, die gar nichts abholt.

⚠ **TLS über Fingerabdruck, nicht über eine PKI.** In einer Kommune gibt es selten eine Zertifikatsverwaltung. Beide Seiten tauschen deshalb bei der Registrierung den SHA-256-Abdruck ihres Zertifikats und prüfen künftig genau diesen. Das Zertifikat legt jeder Dienst selbst an – es ist nichts zu beantragen und nichts einzutragen.

6 Was ausfällt, wenn Ebene 0 ausfällt

Ein Administrator sollte wissen, was ein Ausfall bedeutet – und was **nicht**.

Fällt aus	Folge	Keine Folge
Ordvis Service	Neue Post wird nicht abgeholt und nicht zugestellt. Ereignisse der Anwendungen sammeln sich in deren Outbox und gehen nicht verloren .	Die Fachanwendungen laufen normal weiter. Vorhandene Vorgänge lassen sich bearbeiten.
Ordvis Updater	Es lassen sich keine Updates einspielen.	Der laufende Betrieb ist nicht betroffen.
PostgreSQL	Nichts läuft mehr.	–

★ **Ereignisse gehen bei einem Ausfall nicht verloren.** Jede Anwendung schreibt sie in eine eigene Ablage (Outbox) und schickt sie nach, sobald der Dienst wieder erreichbar ist. Das ist der Grund, warum eine Anwendung Ereignisse **niemals** direkt per HTTP schickt.



TEIL II · EINRICHTUNG

Ordivis Service einrichten

Kennwort, Postfächer, Apps und die Reihenfolge, in der es sicher ist

Produktversion 2026.09.07 · Handbuch-Revision 1.0

Stand 07.09.2026 · Plattform .NET 10 / Windows Server

OS-10 Einrichten

Ordvis Service einrichten

Dieses Kapitel führt durch die Erstinbetriebnahme. Es setzt voraus, dass der Dienst installiert ist und läuft; der Aufbau der Anlage steht in *Der Ordvis Stack* (OS-01).

⚠ **Die Reihenfolge in Abschnitt 4 ist keine Empfehlung.** Wer ein Postfach hier einschaltet, bevor er es in der Fachanwendung abschaltet, holt es doppelt ab – und verliert Post.

INHALT

1 Kennwort der Verwaltung

2 Postfächer

2.1 Anmeldung mit OAuth2 (Microsoft 365, Google)

2.2 Bestehende Postfächer aus Ordvis IT übernehmen

3 Apps anbinden

3.1 Nachsehen, ob die Anmeldung wirklich funktioniert

3.2 Ansprüche

4 Die Umschaltung – in dieser Reihenfolge

5 Standardziel und Simulationsbetrieb

1 Kennwort der Verwaltung

```
setx /M Service__AdminPassword "<Kennwort>"
```

Danach **den Dienst neu starten** – eine Umgebungsvariable liest ein laufender Prozess nicht nach.

⚠ **Ohne gesetztes Kennwort ist die Verwaltung nur vom Server selbst erreichbar.** Das ist die sichere Vorgabe, keine Einschränkung: Eine Verwaltung ohne Kennwort im Netz gehört jedem, der sie findet – und über sie lassen sich Postfächer und Zustellziele ändern.

Über unverschlüsseltes HTTP ist sie ebenfalls nur örtlich erreichbar. Aus dem Netz besteht der Dienst auf HTTPS, weil das Kennwort als Kopfzeile über die Leitung geht.

★ **Das Zertifikat legt der Dienst selbst an** – je Installation eines, fünf Jahre gültig, im Speicher `LocalMachine\My`. Es ist nichts zu beantragen und nichts einzutragen. Er hört auf dem Port über dem HTTP-Port: **5310 → 5311**.

2 Postfächer

`Verwaltung → Postfächer → Neu`. Gebraucht werden Server, Port, Sicherheit, Anmeldeverfahren und die Absenderadresse.

★ **Vor dem Speichern den Verbindungstest benutzen.** Er baut die Verbindung wirklich auf, meldet sich an und öffnet den Ordner **nur lesend** – und schreibt jeden Schritt mit. Ein Postfach, das erst beim ersten Abruf scheitert, scheitert stumm.

2.1 Anmeldung mit OAuth2 (Microsoft 365, Google)

Für Postfächer ohne Kennwortanmeldung führt die Verwaltung durch den **Gerätecode**: Der Dienst zeigt einen Code, der Administrator bestätigt ihn im Browser des Anbieters. Ein Kennwort wird dabei nie eingegeben und nie gespeichert.

⚠ Die Zustimmung läuft ab. Der Dienst erneuert das Token selbsttätig, solange er läuft; nach einer längeren Abschaltung kann eine erneute Bestätigung nötig sein.

2.2 Bestehende Postfächer aus Ordvis IT übernehmen

Postfächer → Aus Ordvis IT übernehmen liest die dort eingerichteten Konten über die Schnittstelle und legt sie hier an. Der Import zeigt zuerst eine **Vorschau**.

⚠ **Übernommene Postfächer sind zunächst abgeschaltet, und das ist Absicht.** Wären sie sofort aktiv, holten zwei Systeme dasselbe Postfach ab – ein Wettlauf, bei dem der Zweite die Mail nie sieht. Erst in Ordvis IT abschalten, dann hier einschalten (Abschnitt 4).

⚠ **Kennwörter werden nicht mit übernommen.** Sie liegen in Ordvis IT verschlüsselt und lassen sich dort nicht auslesen. Sie müssen hier einmal eingetragen werden.

3 Apps anbinden

Eine Fachanwendung meldet sich **selbst** an; sie wird nicht von Hand eingetragen. Was der Administrator tut: das **Bootstrap-Token** setzen, das die *erste* Anmeldung erlaubt.

Verwaltung → Einstellungen → Bootstrap-Token. Speichern genügt – **kein Neustart**, die Erstanmeldung liest den Wert bei jedem Versuch neu.

★ **Ein Token für alle Erstanmeldungen, nicht eines je Anwendung.** Es beglaubigt die erste Anmeldung einer App, die der Dienst noch *nicht* kennt. Für eine bereits registrierte App wird es **nicht** mehr angenommen – sie muss ihr eigenes Token vorlegen. Wer eine zweite Anwendung anbindet, braucht deshalb kein neues Token, sondern dasselbe.

⚠ **Ein leeres Feld heißt „hier nicht gesetzt“, nicht „abgeschaltet“.** Dann gilt weiter, was in **appsettings.json** unter **Service:BootstrapToken** steht. Beides leer heißt: Es gibt keine Erstanmeldung – eine offene Registrierung wäre ein Weg, sich eine App-Identität zu verschaffen und damit fremde Zustellungen abzugreifen.

⚠ **In der Schlüsselliste unter dem Formular steht der Wert nicht.** Dort erscheint nur, *dass* er gesetzt ist und wie lang er ist. Zum Kopieren steht er im Feld darüber.

– Die Token werden **GENAU EINMAL** ausgegeben – und zwar an die Anwendung, nicht an Sie. Nach der ersten Anmeldung gibt Ordvis Service der Anwendung zwei eigene Token. Sie gehören in deren Konfiguration, und das Bootstrap-Token wird **aus der Konfiguration der Anwendung** entfernt: Es wäre sonst ein zweiter Weg zu dieser App-Identität. Im Dienst bleibt es stehen – dort wird es für die nächste Anwendung gebraucht.

Schreibt die Anwendung ihr Protokoll auf die Konsole und läuft als Windows-Dienst, ist diese Meldung im Moment ihrer Entstehung verloren. **Prüfen Sie deshalb nach der ersten Anmeldung Abschnitt 3.1.**

3.1 Nachsehen, ob die Anmeldung wirklich funktioniert

Verwaltung → Apps , Spalte **Zustand**.

Anzeige	Bedeutung
ok	Letzte Gesundheitsprüfung erfolgreich
Anmeldung scheitert	⚠ Jemand meldet sich unter dieser Kennung mit einem ungültigen Token an – fast immer ein verbrauchtes Bootstrap-Token
Fehler	Die Gesundheitsprüfung erreicht die Anwendung nicht

⚠ »Anmeldung scheitert« wiegt schwerer als »ok«, und die Reihenfolge in der Anzeige ist **Absicht**. Die Gesundheitsprüfung ist ein Anruf *des Dienstes* bei der Anwendung – sie sagt nichts darüber, ob die Anwendung zurückkommt. Sie stand schon einmal stundenlang auf grün, während dieselbe Anwendung sich bei jedem Start vergeblich anmeldete und deshalb nichts melden, keine Ansprüche eintragen und nichts einliefern konnte.

Sind die Token verloren: Apps → Token **erneuern** . Beide werden angezeigt, jedes nur einmal.

⚠ Die Erneuerung tauscht **BEIDE** Token – auch das, an dem die funktionierende Zustellung hängt. Wer nur eines einträgt, bricht die Richtung, die vorher lief.

3.2 Ansprüche

Eine Anwendung meldet bei der Anmeldung, wofür sie zuständig ist: Postfächer, Objekttypen, Betreffpräfixe, Kategorien. Daraus entstehen die Routing-Regeln der Schicht L2.

Anspruch	Wirkung
Postfächer	Meldung aus diesem Postfach → diese Anwendung
Objekttypen	Ein Objekttyp gehört genau einer Anwendung
Betreffpräfixe	⚠ erzeugt Betreff enthält ...
Kategorien	Kategorie → Anwendung

⚠ **Betreffpräfixe sind gefährlicher, als sie aussehen.** Der Anspruch prüft, ob der Betreff die Zeichenkette **enthält** – nicht, ob er damit beginnt. Ein Präfix wie **IT** trifft damit »Mitteilung«, »Bitte um Rückruf« und »Sicherheit«. Im Zweifel leer lassen.

4 Die Umschaltung – in dieser Reihenfolge

Solange die Fachanwendung ihre Post noch selbst abholt, darf Ordvis Service dasselbe Postfach nicht abholen.

1. **In der Fachanwendung** das Empfangs-Postfach deaktivieren. ⚠ Die **SMTP**-Konten bleiben aktiv, wenn die Anwendung ihre Post weiterhin selbst versendet.
2. **Prüfen**, dass dort seither nichts mehr abgeholt wird.
3. **In Ordvis Service** das Postfach einschalten.
4. Eine Testmail schicken und in **Verwaltung → Eingang** nachsehen, dass sie ankommt.
5. In der Fachanwendung nachsehen, dass daraus ein Vorgang wurde.

⚠ **Nie umgekehrt.** Beide gleichzeitig aktiv heißt: Wer zuerst kommt, nimmt die Mail. Der andere sieht sie nie, und es sieht nach einem Fehler in der Anwendung aus.

5 Standardziel und Simulationsbetrieb

Standardziel (`Service:DefaultTargetApp`) ist die Anwendung, an die eine Meldung geht, für die keine Regel greift. Ohne Standardziel bleibt sie liegen.

Simulationsbetrieb (`Service:SimulationOnly`) lässt den Dienst alles tun außer zustellen – er holt ab, routet, protokolliert, stellt aber nichts zu.

★ **Das ist der richtige Zustand für die ersten Tage.** In **Verwaltung → Eingang** lässt sich dann nachsehen, welche Meldung wohin gegangen *wäre*, ohne dass ein Vorgang entsteht.

⚠ **Und der richtige Moment, ihn abzuschalten, ist eine bewusste Entscheidung.** Danach wird jede eingehende Mail zu einem Vorgang.

Betrieb und Störungssuche

Prüfen, ob es läuft – und finden, woran es liegt

Produktversion 2026.09.07 · Handbuch-Revision 1.0

Stand 07.09.2026 · Plattform .NET 10 / Windows Server

OS-40 Betrieb

Betrieb und Störungssuche

Für den laufenden Betrieb und den Moment, in dem etwas nicht ankommt. Aufgebaut in der Reihenfolge, in der man vorgeht: erst sehen, ob es läuft, dann suchen, wo es hängt.

INHALT

1 Prüfen, ob es läuft

1.1 Die Übersicht

1.2 Messgrößen im laufenden Betrieb

2 Es kommt keine Mail an

3 Die Mail ist da, aber es entsteht kein Vorgang

4 Eine Anwendung meldet nichts mehr

5 Eine Meldung ging an die falsche Anwendung

6 Sicherung und Aktualisierung

7 Wenn gar nichts mehr geht

1 Prüfen, ob es läuft

1.1 Die Übersicht

Verwaltung → Übersicht zeigt Eingang, Zustellungen, Nachrichten und die Trefferverteilung der Routing-Schichten.

★ Die Trefferverteilung ist die aussagekräftigste Zahl auf dieser Seite. Eine Anlage, bei der alles auf L4 trifft, hat keine Regeln – sie stellt an das Standardziel zu, *weil* es das Standardziel ist, nicht weil es zuständig wäre. Das funktioniert, solange es nur eine Anwendung gibt, und fällt in dem Moment auf die Füße, in dem eine zweite dazukommt.

Schicht	Was sie entscheidet
L1	Der Vorgang ist bekannt (Betreffmarke) → an dieselbe Anwendung wie beim ersten Mal
L2	Ansprüche der Anwendungen: Postfach, Objekttyp, Präfix, Kategorie
L3	Regeln des Betreibers
L4	Standardziel

1.2 Messgrößen im laufenden Betrieb

Ohne zusätzliche Software, auf dem Server:

```
dotnet-counters monitor --process-id (Get-CimInstance Win32_Service -Filter "Name='Ordvis.S
```

Messgröße	Sagt
<code>ordivis.umschlaege</code>	Eingegangene Meldungen nach Ausgang
<code>ordivis.zustellungen</code>	Zustellversuche nach Ergebnis und Ziel-App
<code>ordivis.zustelldauer</code>	Dauer je Versuch – zeigt, welche eine Anwendung nicht antwortet
<code>ordivis.nachrichten</code>	Ausgehende Meldungen

⚠ **ohne-empfaenger** ist kein Fehler. Es ist der Normalfall, wenn eine Anwendung ihre Post an den Melder selbst verschickt: Das Ereignis ist dann ein Nachweis, keine Benachrichtigung.

2 Es kommt keine Mail an

Der Reihe nach – die erste Frage ist fast immer die richtige:

1. **Holt der Dienst überhaupt ab?** `Verwaltung` → `Postfächer` : Steht dort ein Zeitpunkt beim letzten Abruf, und steht ein Fehler dabei? Den Verbindungstest laufen lassen – er nennt den Schritt, an dem es endet.
2. **Holt jemand anders ab?** Ein Postfach, das in der Fachanwendung noch aktiv ist, wird doppelt abgeholt.
⚠ **Wer zuerst kommt, nimmt die Mail** – der andere sieht sie nie und meldet keinen Fehler.
3. **Wurde die Meldung aussortiert?** `Verwaltung` → `Eingang` zeigt auch die gefilterten mit Grund: Abwesenheitsnotiz, Bounce, Absenderflut.

3 Die Mail ist da, aber es entsteht kein Vorgang

`Verwaltung` → `Zustellungen` . Der Zustand sagt, was passiert ist:

Zustand	Bedeutung	Was zu tun ist
offen	noch nicht versucht oder wird wiederholt	abwarten
zugestellt	Die Anwendung hat einen Vorgang angelegt – die Kennung steht daneben	nichts
abgewiesen	⚠ Kein Fehler. Die Anwendung hat bewusst keinen Vorgang angelegt: Abwesenheitsnotiz, Bounce oder Dublette	nichts – siehe unten
aufgegeben	Die Versuche sind erschöpft; der Grund steht dabei	Grund lesen, Ursache beheben, erneut zustellen

⚠ **»Abgewiesen« wird bewusst nicht wiederholt.** Eine Abwesenheitsnotiz, die als Störung gewertet würde, ließe den Dienst endlos zustellen. Die Meldung ist nicht verloren – sie steht als Umschlag im Eingang.

4 Eine Anwendung meldet nichts mehr

Erst den **Zustand** prüfen (OS-10, Abschnitt 3.1). Steht dort *Anmeldung scheitert*, ist das die Ursache – nicht die Anwendung.

Ist die Anmeldung in Ordnung, liegt es meist an der Anwendung selbst: Ihre Outbox hält, was sie nicht loswird.

⚠ Ein **422** dort ist ein **Vertragsfehler und wird bewusst nicht wiederholt**. Die Meldung nennt das fehlende Feld. Ein Beispiel aus der Praxis: Eine Anwendung meldete Ereignisse unter einer Vorgangskennung, deren Präfix eine *andere* Anwendung als ihren Nummernkreis registriert hatte.

5 Eine Meldung ging an die falsche Anwendung

Verwaltung → Eingang → Simulieren zeigt den **vollständigen Trefferpfad**: welche Schicht, welche Regel, welches Ziel – ohne etwas zu ändern.

★ Damit findet man die Regel, die zu früh greift, statt zu raten. Danach **neu routen** – für einen Umschlag oder für alle offenen.

6 Sicherung und Aktualisierung

Beides läuft über **Ordvis Updater**. Der **pre**-Hook sichert die Datenbank vor jedem Update, und die Sicherung wird **als Archiv geprüft**, nicht nur geschrieben.

⚠ **Was ein Update nie anfasst:** `appsettings*.json`, Schlüsseldateien, `data/`, `server-url.txt`, `*.db`. Dort stehen Token, Anbindung und maschinenspezifische Werte. Ein Update, das sie ersetzt, nimmt der Anlage ihre Identität: Sie läuft danach und gehört niemandem mehr.

Ausführlich im **Administratorhandbuch von Ordvis Updater**.

7 Wenn gar nichts mehr geht

Der Dienst ist **nicht** die Voraussetzung dafür, dass die Fachanwendungen laufen.

Fällt aus	Folge	Keine Folge
Ordvis Service	Keine neue Post wird abgeholt oder zugestellt. Ereignisse sammeln sich in der Outbox der Anwendungen und gehen nicht verloren .	Die Anwendungen laufen weiter, Vorgänge lassen sich bearbeiten.

★ Der Rückweg ist immer derselbe **Schalter, mit dem umgestellt wurde**: In der Fachanwendung das Empfangs-Postfach wieder aktivieren, in Ordvis Service abschalten. Erst das eine, dann das andere — nie beide gleichzeitig.