Ordivis

GESAMTDOKUMENTATION

Ordivis Platform – Handbuch-Übersicht

Benutzer- und Administrationshandbuch – Überblick, Impressum und Dokumentenkarte

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-00 Impressum

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Ordivis Platform – Handbuch-Übersicht

Willkommen zur vollständigen Dokumentation von **Ordivis Platform**, der integrierten Plattform für IT-Service-Management (ITSM) und IT-Dokumentation. Dieses Handbuchwerk ist **modular** aufgebaut: Jedes Modul und jede Zielgruppe besitzt ein **eigenes PDF-Dokument**. Alle Einzeldokumente teilen sich Layout, Corporate Identity und Nummerierung und lassen sich zu einem **Gesamt-PDF** zusammenfügen (`00_Ordivis-Handbuch-Gesamt.pdf`).

INHALT

So ist die Dokumentation aufgebaut
Dokumentenkarte
- Grundlagen
- Für Anwender
- Fachmodule
- Administration
- Betrieb
- Technische Referenzen und Anhang

Über dieses Handbuch
Wie Sie etwas finden
Konventionen und Symbole
- Die acht Symbole
- So ist jede Handlungsanweisung aufgebaut

Dokumentinformationen
Urheberrecht
Verwandte Themen

So ist die Dokumentation aufgebaut

Sie müssen dieses Handbuchwerk nicht am Stück lesen. Suchen Sie zuerst Ihre **Rolle** in der folgenden Tabelle – sie nennt die Dokumente, die für Ihre Aufgaben zählen. Alles Weitere schlagen Sie über die Dokumentenkarte darunter nach.

Rolle	Was Sie tun	Ihr Einstieg	Weiterführend
Endbenutzer / Mitarbeiter	Anliegen melden, Status verfolgen, Wissen lesen	ID-10	ID-12, ID-25, ID-34
Service-Desk-Agent	Tickets bearbeiten, kommunizieren, Wissen pflegen	ID-11	ID-20, ID-25, ID-34
Fachanwender	CMDB, IPAM, Assets, Netze und Abläufe pflegen	ID-21	ID-22 … ID-27a, ID-29a, ID-33
Administrator	Benutzer, Rollen, Mandanten, E-Mail, Lizenz konfigurieren	ID-40	ID-41 … ID-44, ID-31

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Rolle	Was Sie tun	Ihr Einstieg	Weiterführend
IT-Betrieb / Server-Admin	Installation, Update, Sicherung, Härtung	ID-50	ID-02, ID-51
Informationssicherheit (ISB)	ISMS führen, Risiken und Maßnahmen nachweisen	ID-28	ID-28a, ID-44
Datenschutz (DSB)	Verarbeitungstätigkeiten führen, Betroffenenrechte	ID-32	ID-27, ID-44
Notfallmanagement (BCB)	Fortführung planen, Wiederanlauf üben	ID-29	ID-30, ID-29a
Entwickler / Integration	Fremdsysteme anbinden, Daten auswerten	ID-60	ID-52, ID-27a
Revision / Audit	Nachweise und Berechtigungen prüfen	ID-44	ID-27, ID-28a, ID-32

Dokumentenkarte

Die Karte listet **alle** Einzeldokumente in der Reihenfolge des Gesamt-PDF. Dokumente mit Buchstabenzusatz (z. B. ID-21a) vertiefen das Dokument mit derselben Nummer.

Grundlagen

Code	Dokument	Zielgruppe	Wofür
ID-00	Handbuch-Übersicht & Impressum	Alle	Navigation, Konventionen, Impressum
ID-01	Einführung & Grundlagen	Alle	Begriffe, Oberfläche, erste Schritte
ID-02	Systemvoraussetzungen & Architektur	Betrieb / Admin	Komponenten, Hardware, Netzwerk

Für Anwender

Code	Dokument	Zielgruppe	Wofür
ID-10	Benutzerhandbuch – Self-Service-Portal	Endbenutzer	Anliegen melden und verfolgen
ID-11	Agentenhandbuch – Service Desk	Agenten	Tickets bearbeiten
ID-12	Darstellung und Barrierefreiheit	Alle	Farbschema, Kontrast, Schriftgröße

Fachmodule

Code	Dokument	Zielgruppe	Wofür
ID-20	Modul Service Desk / Ticketing	Agenten / Admin	Prozesse, SLA, Kategorien, Routing
ID-20a	Monitoring-Anbindung	Admin / Betrieb	Alarme als Vorgänge, Entwarnung, Wartungsfenster

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Code	Dokument	Zielgruppe	Wofür
ID-21	Modul CMDB (Configuration Management)	Fachanwender	CIs, Klassen, Beziehungen
ID-21a	Datenimport – Geschäftsprozesse	Fachanwender / Admin	Prozesse per CSV/JSON übernehmen
ID-21b	Benutzerkonten in der CMDB	Admin / Fachanwender	Konten als CI führen
ID-21c	Rechenzentrums-Dokumentation	Betrieb / Fachanwender	Racks, Komponenten, Patchfelder, Wegverfolgung
ID-22	Modul IPAM (IP-Adressverwaltung)	Fachanwender	Subnetze, Adressen, DHCP
ID-23	Modul Asset- & Vertragsmanagement	Fachanwender	Geräte, Verträge, Lizenzen
ID-24	Modul Discovery (Netzwerk-Inventarisierung)	Fachanwender / Admin	Scans per WMI, SSH, SNMP, LDAP
ID-25	Modul Wissensdatenbank	Agenten / Endbenutzer	Artikel, Freigabe, KEDB
ID-26	Modul Workflows & Aufgaben	Fachanwender	Abläufe und Genehmigungen
ID-27	Modul Reporting & Dashboards	Alle	Kennzahlen, Berichte, Audit-Log
ID-27a	Reports & SQL-Abfragen	Fachanwender / Reporting	Report-Baukasten und freies SQL
ID-28	Modul Informationssicherheit (ISMS: BSI & ISO 27001)	ISB / Admin	Schutzbedarf, Risiken, Maßnahmen
ID-28a	Anlage: ISMS-Dokumentation nach ISO/IEC 27001	ISB	Gelenkte Dokumente und Vorlagen
ID-29	Modul Business Continuity Management (BCM)	BCB / Admin	BIA, Fortführungspläne, Übungen
ID-29a	Modul Checklisten	Fachanwender / Betrieb	Vorlagen mit Durchführungsnachweis
ID-30	Modul Notfall- & Wiederanlaufplanung (DR)	Betrieb	Technischer Wiederanlauf
ID-31	Modul Benachrichtigungen & E-Mail-Integration	Admin	Glocke, Versand, Posteingang
ID-32	Modul Datenschutz (Art. 30 DSGVO)	DSB / Admin	Verarbeitungsverzeichnis, Meldewesen
ID-33	Modul Organisation & Geschäftsverteilungsplan	Fachanwender / Leitung	Aufbau, Abläufe, Produktkatalog
ID-34	Modul Selbstorganisation	Alle	Aufgabenbrett, Aufgaben, Kalender

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Code	Dokument	Zielgruppe	Wofür
ID-35	Modul Datensicherung	Betrieb / Admin	Sicherungen planen, protokollieren, kontrollieren

Administration

Code	Dokument	Zielgruppe	Wofür
ID-40	Administrationshandbuch	Administrator	Ersteinrichtung, Jobs, Branding
ID-41	Modul Identität, Benutzer & Rollen (RBAC)	Administrator	Konten, Rollen, MFA, Passwörter
ID-42	Modul Mandantenfähigkeit (Multi-Tenancy)	Administrator	Mehrere Organisationen trennen
ID-43	Modul Produktlizenzierung & Aktivierung	Administrator	Testphase, Editionen, Limits
ID-44	Rollen- & Rechte-Matrix	Admin / Audit / Fachpersonal	Vollständiger Rechtekatalog

Betrieb

Code	Dokument	Zielgruppe	Wofür
ID-50	Betriebshandbuch – Installation & Serverbetrieb	IT-Betrieb	Installation, Update, Sicherung, Härtung
ID-51	Datenbank-Leistung & Dimensionierung	IT-Betrieb	Verbindungspools, Größenklassen
ID-53	Produktsicherheit & Meldewege	IT-Betrieb	Schwachstellen melden, Updates, Außerbetriebnahme

Technische Referenzen und Anhang

Code	Dokument	Zielgruppe	Wofür
ID-52	Datenbank-Referenz	Entwickler / Reporting	Tabellen und Felder für SQL
ID-60	API-Referenz	Entwickler / Integration	REST-Endpunkte, Authentifizierung
ID-90	Anhang – Glossar & Referenzen	Alle	Glossar, Stichwortverzeichnis, Historie

Über dieses Handbuch

Dieses Handbuch folgt anerkannten Standards für technische Dokumentation (u. a. **DIN EN IEC/IEEE 82079-1** „Erstellen von Nutzungsinformationen“ und **ISO/IEC 26514**). Das bedeutet konkret:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Zielgruppengerecht:** Endbenutzer-Teile sind in einfacher Sprache mit genauen Schritt-für-Schritt-Anleitungen geschrieben; administrative Teile setzen IT-Kenntnisse voraus.
- **Handlungsorientiert:** Jede Funktion folgt dem Muster **Zweck** → **Voraussetzungen** → **Schritt-für-Schritt** → **Hinweise/Stolperfallen**.
- **Konsistent:** Einheitliche Begriffe (siehe Glossar, ID-90), einheitliches Layout im Material-Design-3-Erscheinungsbild der Anwendung.
- **Sicherheitshinweise** sind als farbige Karten hervorgehoben (Abschnitt *Konventionen und Symbole*).

Wie Sie etwas finden

Vier Wege führen zum Ziel, je nachdem, was Sie schon wissen:

Sie wissen …	Dann nutzen Sie …
Ihre Rolle	die Rollentabelle am Anfang dieses Dokuments
das Modul oder Thema	die Dokumentenkarte, danach das Inhaltsverzeichnis des Dokuments
einen Begriff	das **Stichwortverzeichnis** in ID-90; unbekannte Abkürzungen erklärt das Glossar ebendort
nichts davon	die Volltextsuche im PDF-Betrachter (meist `Strg` + `F`) über das Gesamt-PDF

Jedes Dokument endet mit dem Abschnitt **Verwandte Themen**. Er nennt die Dokumente, die an dieser Stelle typischerweise als Nächstes gebraucht werden.

Konventionen und Symbole

Im gesamten Handbuchwerk gelten folgende Konventionen:

- **Menüpfade** werden mit ▸ angegeben, z. B. *Verwaltung* ▸ *Benutzer und Rollen*.
- **Schaltflächen und Feldnamen** stehen in *Kursivschrift*.
- **Eingaben und technische Werte** stehen in `Festbreitenschrift`.
- Hinweiskarten sind farblich gekennzeichnet:

Hinweis: Allgemeine, hilfreiche Zusatzinformation.

Wichtig / Voraussetzung: Etwas, das erfüllt sein muss, bevor Sie fortfahren.

Achtung: Aktion mit möglicherweise unerwünschten oder nicht umkehrbaren Folgen.

Tipp: Empfehlung aus der Praxis für effizienteres Arbeiten.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Die acht Symbole

Mehr als diese acht Symbole gibt es nicht. Jedes steht immer für dasselbe, in jedem Dokument. Jedes trägt sein Wort mit sich – wer das Zeichen nicht kennt, liest die Bedeutung daneben.

Symbol	Bedeutung	Wo es steht
💻	Der Weg im **Windows-Client**	im Anleitungskopf
🌐	Der Weg im **Web-Portal**	im Anleitungskopf
⚙️	Der Weg **am Server** – Konsole, Konfigurationsdatei, Datenbank	im Anleitungskopf
🔑	Das **Recht**, das die Handlung verlangt	im Anleitungskopf
▸	Trennt die Stationen eines **Menüpfads**	überall
🔒	**Pflichtreihenfolge** – die Schritte bauen aufeinander auf	vor der Schrittfolge
💡	**Empfohlene Reihenfolge** – Abweichen ist möglich, kostet aber Nacharbeit	vor der Schrittfolge
✅	**Ergebnis** – woran Sie erkennen, dass es geklappt hat	nach der Schrittfolge

So ist jede Handlungsanweisung aufgebaut

Jede Handlung, die Sie ausführen können, steht im Handbuch nach demselben Muster. Das gilt für das Melden eines Druckerproblems ebenso wie für das Anlegen eines Mandanten – nur der Inhalt unterscheidet sich, nicht der Aufbau:

1. Der Anleitungskopf sagt zuerst, **wo** die Handlung stattfindet und **was Sie dürfen müssen.** Er steht als blaue Karte unmittelbar unter der Überschrift:

> 💻 **Client:** *Assets & Lizenzen* ▸ *Verträge* · 🌐 **Portal:** *Assets & Lizenzen* ▸ *Verträge* · 🔑 `contracts.manage`

Manches lässt sich in keiner der beiden Oberflächen erledigen, sondern nur unmittelbar am Server – eine Konfigurationsdatei, ein Dienst, die Datenbank. Auch das sagt der Kopf:

> ⚙️ **Am Server:** `appsettings.Production.json`, danach Neustart des Dienstes *InfraDesk.API* · 🔑 Schreibzugriff auf die Konfiguration

Gibt es die Funktion nur in einer Oberfläche, sagt der Kopf das ausdrücklich – und nennt den Grund, statt die andere Oberfläche stillschweigend wegzulassen:

> 💻 **Nur im Client:** *CMDB* ▸ *Gerätemodell-Katalog* · 🔑 `cmdb.class.manage` — Das Portal bietet den Katalog nicht an; im Browser fehlt der Zugriff auf den Etikettendrucker.

2. Voraussetzungen stehen als gelbe Karte davor, wenn es welche gibt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3. Die Schrittfolge ist nummeriert. **Ein Schritt = eine Handlung.** Jeder Schritt nennt das Bedienelement beim Namen, das Sie anklicken, ausfüllen oder auswählen.

4. Die Reihenfolge ist gekennzeichnet, sobald sie zählt – und nur dann. Steht keine Marke da, können Sie die Schritte in beliebiger Reihenfolge ausführen:

🔒 **Pflichtreihenfolge** — die Schritte bauen aufeinander auf; ein Schritt braucht das Ergebnis des vorigen. Wer sie tauscht, findet das nächste Feld nicht oder überschreibt sich selbst.

💡 **Empfohlene Reihenfolge** — es geht auch anders herum, verursacht aber Nacharbeit. Die Karte sagt jedes Mal dazu, welche.

5. Das Ergebnis schließt ab: woran Sie erkennen, dass die Handlung gewirkt hat.

> **Warum so streng.** Eine Anleitung, die in einem Modul aus zehn nummerierten Schritten besteht und im nächsten aus dem Satz „Legen Sie die Schritte fest", ist an der zweiten Stelle keine Anleitung mehr. Sie kostet den Leser genau dort Zeit, wo er sie am wenigsten hat – er kennt das Modul ja noch nicht. Deshalb ist der Aufbau vorgeschrieben und wird beim Bau geprüft (`pruefe_handbuch.py`).

Dokumentinformationen

Angabe	Wert
Produkt	Ordivis Platform – IT-Service-Management & IT-Dokumentation
Dokument	Handbuch-Übersicht & Impressum
Dokument-ID	ID-00
Handbuch-Revision	1.0
Produktbezug	Version 2026.08.19
Plattform	.NET 10 / Windows Server
Stand	19.08.2026
Herausgeber / Urheber	Christian Grams
Ansprechpartner / Support	ausschließlich per E-Mail und Ticket
Reaktionszeiten	kritische Störungen 8 Stunden, sonstige Anfragen 3 Arbeitstage
Änderungshistorie	ID-90, Abschnitt *Versionshistorie des Handbuchs*

Revision und Stand gelten für **alle** Dokumente dieses Handbuchwerks; sie werden beim Bau eingesetzt und stehen zusätzlich auf jeder Titelseite.

Urheberrecht

© 2026 Christian Grams / Ordivis Platform. Alle Rechte vorbehalten.

Dieses Dokument sowie die darin beschriebene Software sind urheberrechtlich geschützt. Die Weitergabe, Vervielfältigung, Speicherung in Datenverarbeitungsanlagen sowie Übersetzung – auch auszugsweise – ist nur mit vorheriger schriftlicher Genehmigung des Urhebers gestattet. Alle genannten Marken- und Produktnamen sind Eigentum ihrer jeweiligen Inhaber und werden ausschließlich zu Identifikationszwecken verwendet.

Die Inhalte dieses Handbuchs wurden mit größter Sorgfalt erstellt. Für die Richtigkeit, Vollständigkeit und Aktualität wird gleichwohl keine Gewähr übernommen. Ordivis Platform befindet sich in aktiver Weiterentwicklung; Abbildungen und Funktionsumfang können je nach Version und Edition abweichen. Maßgeblich für den Funktionsumfang ist die jeweils installierte Produktversion.

Verwandte Themen

- **ID-01 – Einführung & Grundlagen** – der inhaltliche Einstieg für alle Rollen
- **ID-90 – Anhang – Glossar & Referenzen** – Glossar und Stichwortverzeichnis zum Nachschlagen
- **ID-44 – Rollen- & Rechte-Matrix** – welche Rolle was sehen und ändern darf

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL I · GRUNDLAGEN

Einführung & Grundlagen

Was ist Ordivis Platform, wie ist es aufgebaut, welche Begriffe muss ich kennen

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-01 Einführung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Einführung & Grundlagen

Dieses Dokument gibt allen Zielgruppen einen gemeinsamen Einstieg: Was Ordivis Platform leistet, wie es bedient wird und welche Grundbegriffe im gesamten Handbuch verwendet werden. Lesen Sie es einmal, bevor Sie zu den rollenspezifischen Dokumenten wechseln.

INHALT

1 Was ist Ordivis Platform?
2 Die beiden Oberflächen
3 Erste Schritte
3.1 Anmelden
3.2 Die Oberfläche im Überblick
3.3 Wo liegt was? Client und Portal nebeneinander
3.4 Das persönliche Dashboard
4 Grundbegriffe & Konzepte
5 ITIL-Bezug in Kürze
6 Für wen ist welches Dokument?
7 Hilfe direkt in der Anwendung
Verwandte Themen

1 Was ist Ordivis Platform?

Ordivis Platform ist eine **integrierte Plattform für IT-Service-Management (ITSM) und IT-Dokumentation**. Sie vereint Aufgaben, die sonst auf viele Einzelwerkzeuge verteilt sind, in einer Anwendung:

- **Service Desk / Ticketing** – Anfragen, Störungen und Aufträge erfassen und bearbeiten (nach den ITIL-Prozessen Incident, Service Request, Problem, Change).
- **CMDB** – alle IT-Bausteine („Configuration Items“) und ihre Beziehungen dokumentieren.
- **IPAM** – Netze und IP-Adressen verwalten.
- **Asset- & Vertragsmanagement** – Geräte, Verträge und Software-Lizenzen über ihren Lebenszyklus.
- **Wissensdatenbank** – Lösungen als wiederverwendbare Artikel; Selbsthilfe für Endbenutzer.
- **Informationssicherheit (ISMS)** – BSI IT-Grundschutz und ISO/IEC 27001 in einem Modul.
- **Business Continuity (BCM) & Notfallplanung (DR)** – Geschäftsfortführung und Wiederanlauf.
- **Weitere Module** – Workflows, Reporting, Benachrichtigungen, Discovery.

Ordivis Platform ist **mandantenfähig** (mehrere Organisationseinheiten in einer Installation) und **rollenbasiert** (jede Person sieht und darf genau das, was ihrer Rolle entspricht).

> **Hinweis:** Ordivis Platform orientiert sich an **ITIL**. Wo möglich verwendet das Handbuch die dort üblichen Begriffe und erklärt sie für Einsteiger.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2 Die beiden Oberflächen

Ordivis Platform wird über **zwei** Oberflächen genutzt:

1. **Windows-Client** (Desktop-App) – die vollwertige Arbeitsoberfläche für **Agenten, Fachanwender und Administratoren**.
2. **Web-Portal** (im Browser) – seit Version 2026.08 **ebenfalls eine vollwertige Arbeitsoberfläche**, nicht mehr nur Selbstbedienung: Endbenutzer melden hier ihre Anliegen, Agenten bearbeiten Tickets, Probleme und Changes, Fachanwender pflegen CMDB, Assets, ISMS und Datenschutz, Administratoren verwalten Benutzer, Rollen und Mandanten.

> **Merksatz:** Beide Oberflächen sprechen **dieselbe Schnittstelle** und setzen **dieselben Rechte** durch. Was Sie sehen und dürfen, hängt an Ihrer Rolle – nicht daran, welche Oberfläche Sie öffnen.

Was nur der Windows-Client kann. Einige wenige Funktionen hängen am Windows-Arbeitsplatz selbst und lassen sich im Browser nicht anbieten: das Drucken auf Etikettendrucker über die lokale Schnittstelle, das Einlesen von NFC-Etiketten sowie die Netzwerk-Inventarisierung im lokalen Netz. Das Portal weist an der jeweiligen Stelle darauf hin, statt die Funktion stillschweigend wegzulassen.

Die **Windows-Anmeldung ohne Kennworteingabe** stand bis 2026.08 ebenfalls in dieser Liste. Sie ist seit dieser Fassung auch im Portal möglich, wenn Ihre Administration sie einrichtet – siehe Betriebshandbuch 3.4. Ohne Einrichtung bleibt es bei der Anmeldung mit Benutzername und Kennwort.

Was nur das Portal kann. Es ist von jedem Gerät mit Browser erreichbar – auch vom Tablet, vom Mobiltelefon und vom Rechner im Homeoffice – und braucht keine Installation.

3 Erste Schritte

3.1 Anmelden

> 💻 **Client:** Programmstart · 🌐 **Portal:** die Portal-Adresse im Browser · 🔑 kein Recht nötig – die Rechte entscheiden erst danach, was Sie sehen

> **Voraussetzung:** Ein Administrator hat Ihnen ein Konto **und** ein erstes Kennwort vergeben. Neu angelegte lokale Benutzer haben zunächst **kein** Kennwort und kommen ohne diesen Schritt nicht hinein (ID-41).

🔒 **Pflichtreihenfolge** — der Mandant wird erst nach der Kennworteingabe abgefragt, der zweite Faktor erst nach dem Mandanten. Vorher gibt es die Auswahl nicht.

1. Starten Sie den **Ordivis-Client** über das Symbol auf dem Desktop – oder öffnen Sie die **Portal-Adresse** im Browser, die Sie von Ihrer IT erhalten haben.
2. Tippen Sie Ihren **Benutzernamen** in das erste Feld. Groß- und Kleinschreibung spielt hier keine Rolle.
3. Tippen Sie Ihr **Kennwort** in das zweite Feld. Hier zählt die Groß- und Kleinschreibung sehr wohl.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4. Klicken Sie auf *Anmelden*.
5. Gehören Sie zu **mehreren Mandanten**, wählen Sie jetzt den gewünschten Mandanten aus der Liste und bestätigen Sie.
6. Ist **Mehr-Faktor-Authentifizierung (MFA)** eingeschaltet, geben Sie den zweiten Faktor ein – in der Regel den sechsstelligen Code aus Ihrer Authenticator-App (ID-41, Abschnitt 6).

✅ **Ergebnis:** Sie landen auf dem persönlichen Dashboard. Oben rechts steht Ihr Name, bei mehreren Mandanten daneben der gerade gewählte Mandant.

3.2 Die Oberfläche im Überblick

Beide Oberflächen sind gleich aufgebaut. Nach der Anmeldung sehen Sie:

- **Navigationsmenü links** – die Bereiche, jeder aufklappbar. Es ist immer nur *ein* Bereich zugleich geöffnet, damit die Liste kurz bleibt. Über dem Menü steht ein **Suchfeld**: Ein dort eingetippter Begriff lässt nur die passenden Einträge stehen – schneller, als sich zu erinnern, in welchem Bereich etwas liegt.
- **Fußbereich unter dem Menü** – *Hilfe*, Ihr Konto, *Systemstatus* und *Abmelden*.
- **Persönliches Dashboard** (im Client *Start*, im Portal *Übersicht*) – Ihre eigenen Tickets, Aufgaben, Wiedervorlagen, Genehmigungen und Kennzahlen, abhängig von Ihrer Rolle.
- **Inhaltsbereich rechts** – die jeweils geöffnete Liste oder Detailseite.

Sie sehen im Menü **nur die Bereiche, für die Sie ein Recht haben** und deren Modul eingeschaltet ist. Ein leeres Menü ist deshalb kein Fehler, sondern eine Rollenfrage – siehe ID-44.

3.3 Wo liegt was? Client und Portal nebeneinander

Die Bereiche tragen in beiden Oberflächen **dieselben Namen** und stehen in derselben Reihenfolge. Das heißt aber nicht, dass dahinter dieselben Seiten liegen: Der Client ist am Fachbearbeiter gebaut, das Portal zusätzlich am Endbenutzer. Die Tabelle sagt für jeden Bereich, was Sie wo finden.

Bereich	💻 Client	🌐 Portal
Start	*Start*	*Übersicht*
Meine Arbeit (Client) / **Meine Vorgänge** (Portal)	Aufgabenbrett, Meine Aufgaben, Mein Kalender, Meine Workflow-Aufgaben, Playbooks/Checklisten, Wetter und Warnungen	Meine Tickets, Neues Ticket, Service-Katalog, Meine Aufgaben, Mein Aufgabenbrett, Meine Aufgabenliste, Mein Kalender, Genehmigungen, Checklisten, Checklisten-Vorlagen, Datensicherung prüfen
Mein Arbeitsplatz	–	Meine Geräte, Awareness & Bestätigungen, Datenschutz, Verarbeitung melden, Datenschutz-Zentrale, Meine Verfahren
Wissen & Hilfe	(unter *Service Desk*: Wissensdatenbank)	Wissensdatenbank, Wissensdatenbank-Redaktion, Bekannte Fehler (KEDB)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Bereich	💻 Client	🌐 Portal
CMDB	Dashboard, Configuration Items, Standorte, Gerätemodell-Katalog, CI-Hierarchie, Scan-Abgleich, Discovery-Scan, Discovery-Collector, Client-Inventar, Datenqualität, Abhängigkeiten, Business Service Map	CMDB-Dashboard, CMDB-Suche, Standorte und Hierarchie, Abhängigkeits-Karte, Auswirkungsanalyse
Service Desk	Dashboard, Tickets, Probleme, Changes, Wissensdatenbank, Textbausteine, Monitoring-Anbindung	Service-Desk-Vorgänge, Service-Desk-Auswertung, Probleme, Changes, Change-Kalender, Kontakte, Datenqualität, Regelwerk, Zuständigkeit, Textbausteine und Vorlagen, Lageübersicht
Prozesse	Dashboard, Prozessübersicht, Prozessdokumentation	Prozessübersicht, Prozess-Editor
Organisation	Ablauforganisation, Aufbauorganisation, Funktionendiagramm, Organisationshandbuch, Produktkatalog, Geschäftsverteilungsplan, OZG-Leistungen, OZG-Digitalisierungsstand	Aufbauorganisation
Netz (IPAM)	Dashboard, IP-Adressen	Netz & Discovery, Adressverwaltung, Discovery
Assets & Lizenzen	Assets, Lizenzen, Lizenz-Pools, Erkannte Software, Verträge	Assets, Lizenzen, Verträge, Lieferanten, Inventur
Informationssicherheit	ISMS-Dashboard, Sicherheitskatalog, Informationsverbund, Grundschutz-Check, Risikoanalyse, Erklärung zur Anwendbarkeit (SoA), ISMS-Managementsystem	ISMS-Dashboard, ISMS-Arbeitsplatz
Datensicherung	Überblick, Sicherungsläufe, Prüfungen, Pläne & Ziele	nur *Datensicherung prüfen* (unter *Meine Vorgänge*)
Notfallmanagement	Übersicht, BCM-Verbünde, Business Impact Analyse, Bewältigungsorganisation, Alarmierung, Notfalldokumente, Berichte/Übungen/CAPA, Handbücher, Stabs-Portal, DR-Planung	Notfalldokumente, Alarmierungsliste, BCM-Arbeitsplatz, DR-Pläne
Datenschutz	Verarbeitungsverzeichnis, Datenpannen (Art. 33/34), AVV-Register, Betroffenenrechte	im Bereich *Mein Arbeitsplatz* (Zeile 3 dieser Tabelle)
Compliance & Reporting	Audit-Log, Report-Verwaltung	Audit-Log, Betriebs-Dashboard, Berichte, Bericht-Baukasten, Betriebssicht

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Bereich	💻 Client	🌐 Portal
Verwaltung	über den Fußbereich *Einstellungen*	Benutzer und Rollen, Mandanten und Module, Service-Desk-Konfiguration, CI-Klassen, Etiketten
Fußbereich	Hilfe, Einstellungen, Systemstatus, Abmelden	Hilfe, Mein Konto, Benachrichtigungs-Einstellungen, Genehmigungs-Vertretung, Wetter, Systemstatus, Abmelden

Drei Dinge lesen Sie aus der Tabelle heraus:

- **Zwei Bereiche gibt es nur im Portal** – *Meine Vorgänge* und *Mein Arbeitsplatz*. Sie richten sich an alle Mitarbeitenden; der Client ist kein Werkzeug für Endanwender.
- **Die Tiefe liegt im Client.** CMDB, Notfallmanagement, Informationssicherheit und Organisation führen dort deutlich mehr Einträge. Wer diese Module *pflegt*, arbeitet im Client; wer sie *liest*, kommt im Portal weit genug.
- **Keine Oberfläche enthält die andere.** *Lieferanten* und *Inventur* gibt es nur im Portal, *Lizenz-Pools* und *Erkannte Software* nur im Client. Deshalb nennt jede Anleitung in diesem Handbuch beide Wege – und sagt, wenn es nur einen gibt.

> **Hinweis:** Die Namen in dieser Tabelle sind die Namen auf dem Bildschirm. Bis August 2026 hießen vier Bereiche in den beiden Oberflächen verschieden – *ITSM* gegen *Service Desk*, *IPAM* gegen *Netz (IPAM)* und zwei weitere. Sie sind angeglichen; eine ältere Fassung zeigt an diesen Stellen noch die alten Namen.

3.4 Das persönliche Dashboard

Das Dashboard nach der Anmeldung ist **rollenabhängig**: Ein Agent sieht seine offenen Tickets und Fälligkeiten, eine Führungskraft eher Kennzahlen, ein Endbenutzer seine gemeldeten Anliegen.

4 Grundbegriffe & Konzepte

Begriff	Bedeutung (einfach erklärt)
Mandant	Eine abgegrenzte Organisationseinheit/Kunde. Daten verschiedener Mandanten sind getrennt.
Ticket	Ein erfasster Vorgang: Störung (Incident), Auftrag (Service Request), Problem oder Änderung (Change).
Configuration Item (CI)	Ein dokumentierter IT-Baustein in der CMDB (z. B. Server, PC, Dienst, Standort).
Asset	Ein Vermögenswert (Gerät, Vertrag, Lizenz) mit Kosten-/Lebenszyklus-Sicht. Ein Objekt kann zugleich Asset **und** CI sein.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Begriff	Bedeutung (einfach erklärt)
Rolle / Recht	Ihre Rolle bestimmt über **Berechtigungen**, welche Bereiche und Aktionen Sie sehen und ausführen dürfen.
SLA	„Service Level Agreement" – vereinbarte Reaktions- und Lösungszeiten für Tickets.
Priorität	Ergibt sich aus **Dringlichkeit × Auswirkung** (z. B. Hoch × Hoch → Kritisch).
Dringlichkeit / Auswirkung	Eingabewerte am Ticket (je Niedrig / Mittel / Hoch). „Kritisch" ist **keine** Eingabe, sondern eine *abgeleitete Priorität*.

5 ITIL-Bezug in Kürze

Ordivis Platform bildet die vier zentralen Service-Management-Prozesse ab:

- **Incident Management** – Störungen schnell beheben, Betrieb wiederherstellen.
- **Service Request Management** – Standardleistungen (z. B. neue Software) mit Genehmigungen.
- **Problem Management** – Ursachen wiederkehrender Störungen finden (Known Error).
- **Change Management** – Änderungen geplant und genehmigt durchführen (CAB).

Zusätzlich unterstützt die CMDB den ITIL-Prozess **Service Asset & Configuration Management (SACM)**.

6 Für wen ist welches Dokument?

- **Nur Anliegen melden?** → Benutzerhandbuch (ID-10).
- **Tickets bearbeiten?** → Agentenhandbuch (ID-11) + Modul Service Desk (ID-20).
- **CMDB/IPAM/Assets pflegen?** → die jeweiligen Modul-Dokumente (ID-21 ff.).
- **System verwalten?** → Administrationshandbuch (ID-40) und Modul-Dokumente ID-41 bis ID-43.
- **Server installieren/betreiben?** → Betriebshandbuch (ID-50) und Architektur (ID-02).

7 Hilfe direkt in der Anwendung

Dieses Handbuch steht auch **in der Anwendung selbst** zur Verfügung – im Client wie im Portal unter **Hilfe** ganz unten im Menü. Es ist aus denselben Quellen gebaut wie das gedruckte Handbuch und wird mit dem Produkt ausgeliefert. Daraus folgt zweierlei:

- Die Hilfe ist **immer auf demselben Stand wie Ihre Installation** – sie kann nicht veralten, weil sie zum Produkt gehört.
- Sie funktioniert **ohne Internetverbindung**. Gerade im Notfall, wenn die Verbindung nach außen fehlt, bleibt die Dokumentation erreichbar.

Verwandte Themen

- **ID-02 – Systemvoraussetzungen & Architektur** – was der Server und der Arbeitsplatz mitbringen müssen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-10 – Benutzerhandbuch – Self-Service-Portal** – der erste Weg für Endbenutzer
- **ID-11 – Agentenhandbuch – Service Desk** – der erste Weg für Service-Desk-Agenten
- **ID-12 – Darstellung und Barrierefreiheit** – Farbschema, Kontrast und Schriftgröße anpassen
- **ID-90 – Anhang – Glossar & Referenzen** – unbekannte Abkürzungen nachschlagen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL I · TECHNIK

Systemvoraussetzungen & Architektur

Architektur, Komponenten und Voraussetzungen für Betrieb und Client

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-02 Architektur

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Systemvoraussetzungen & Architektur

Dieses Dokument beschreibt den technischen Aufbau von Ordivis Platform und die Voraussetzungen für Server und Arbeitsplätze. Es richtet sich an **IT-Betrieb** und **Administratoren**. Die konkreten Installationsschritte stehen im Betriebshandbuch (ID-50).

INHALT

1 Architektur im Überblick
2 Software-Plattform
3 Voraussetzungen – Server
4 Voraussetzungen – Arbeitsplatz (Client)
5 Netzwerk & Ports
5.1 Dünner Client – standortübergreifend, VPN, weltweit
6 Datensicherheit auf einen Blick
7 Editionen & Lizenzierung
Verwandte Themen

1 Architektur im Überblick

Ordivis Platform läuft **nativ auf Windows Server** – **ohne Docker/Container**. Die Plattform besteht aus mehreren Komponenten, die als **Windows-Dienste** betrieben werden:

Komponente	Aufgabe	Betriebsart
Ordivis API	Zentrale Geschäftslogik, REST-Schnittstelle, Authentifizierung, RBAC	Windows-Dienst (Kestrel/HTTPS)
Discovery-Collector	Netzwerk-Inventarisierung und Scans in einem **entfernten Netz** – optional, eigenes Installationspaket	Windows-Dienst auf dem Rechner im Zielnetz
Web-Portal	Blazor-Web-Portal – Selbstbedienung für Endbenutzer **und** vollwertiger Arbeitsplatz für Agenten, Fachanwender und Administratoren (Port 8080)	Windows-Dienst
PostgreSQL	Datenhaltung (relationale Datenbank)	Nativer Dienst
WinUI-Client	Desktop-Arbeitsoberfläche (unpackaged)	Desktop-App je Arbeitsplatz

Abbildung 1 zeigt, wie diese Komponenten zusammenhängen und in welche Richtung die Verbindungen aufgebaut werden.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

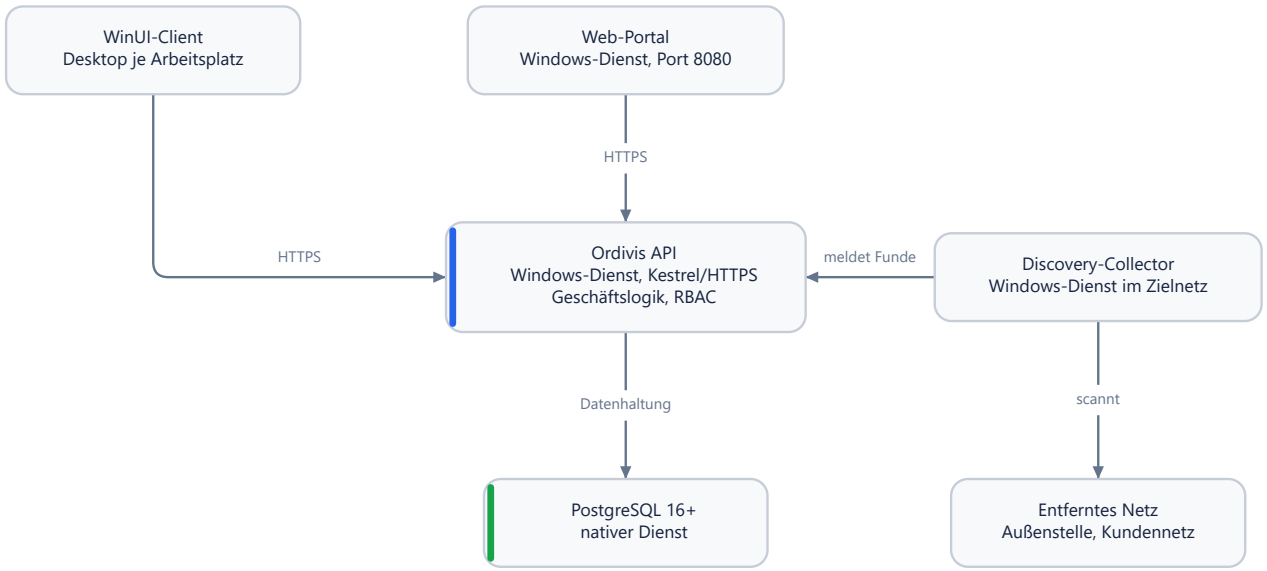

Abb. 1 – Komponenten und ihre Verbindungen. Der Collector im entfernten Netz ruft in der Zentrale an, nicht umgekehrt.

Der modulare Aufbau folgt einer **Clean-Architecture** je Modul (Domain, Application, Infrastructure, Api). Alle Module teilen sich eine Datenbank; die Trennung erfolgt logisch über Mandanten und Berechtigungen.

2 Software-Plattform

- **Laufzeit:** .NET 10 (alle Server-Komponenten und der Client).
- **Datenbank:** PostgreSQL 16 oder neuer (empfohlen 18).
- **Betriebssystem Server:** Windows Server 2022 / 2025.
- **Client-Betriebssystem:** Windows 11 (empfohlen). Windows 10 (ab 19041) ist technisch lauffähig, wird aber **nicht empfohlen** (siehe EOL-Hinweis unten).

3 Voraussetzungen – Server

Ressource	Minimum	Empfohlen
CPU	4 Kerne	8 Kerne
Arbeitsspeicher	8 GB	16 GB+
Datenträger	50 GB SSD	100 GB+ SSD (getrennt für DB)
Netzwerk	1 GbE, fester Hostname (FQDN)	1 GbE, DNS + gültiges TLS-Zertifikat
Zusätzlich	VC++-Runtime (für PostgreSQL-Installer)	Interne Root-CA im Vertrauensspeicher

Voraussetzung: Der PostgreSQL-Installer benötigt die Option `--install_runtimes 1`, sonst scheitert die Installation auf frischen Servern an fehlender VC++-Runtime.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Wichtig – statische MAC-Adresse (Lizenzbindung): Die Produktaktivierung bindet die Lizenz an einen Hardware-Fingerprint, der u. a. die MAC-Adresse der primären Netzwerkkarte einbezieht. In **virtuellen Maschinen** muss die MAC-Adresse der primären vNIC daher **statisch** konfiguriert sein (feste/„static" MAC in Hyper-V, VMware, Proxmox o. ä.). Eine **dynamische MAC**, die sich beim Neustart oder bei einer Migration ändert, führt sonst zu einer fälschlichen Aktivierungssperre („Manipulation erkannt"). Reine Host-Adapter (Hyper-V Default Switch, VPN/TAP, WSL/Docker-Bridges) werden nicht herangezogen und dürfen wechseln. Details siehe ID-43.

4 Voraussetzungen – Arbeitsplatz (Client)

Ressource	Wert
Betriebssystem	**Windows 11** (empfohlen); Windows 10 (19041+) technisch lauffähig, aber EOL
Arbeitsspeicher	4 GB+
Bildschirm	1440×900 oder höher empfohlen
Netzwerk	HTTPS-Erreichbarkeit der API
Zusätzlich	WebView2-Runtime (für grafische BPMN-Darstellung)

Das Self-Service-Portal benötigt am Endbenutzer-Arbeitsplatz nur einen **aktuellen Browser** (Edge, Chrome, Firefox) – keine Installation.

Achtung – Betriebssystem-Lebenszyklus (EOL): Betriebssysteme haben ein Support-Ende (End of Life). **Windows 10 hat am 14. Oktober 2025 sein End-of-Support erreicht** und erhält keine Sicherheitsupdates mehr. Ordivis Platform läuft technisch weiterhin auf Windows 10, der Einsatz wird jedoch **ausdrücklich nicht empfohlen**. Betreiben Sie Clients auf **Windows 11**.

Hinweis: Prüfen Sie auch für **Server** und **PostgreSQL** regelmäßig den Support-Zeitraum. Setzen Sie nur Versionen ein, die noch Sicherheitsupdates erhalten (z. B. Windows Server 2022/2025, aktuell gepflegte PostgreSQL-Hauptversionen). EOL-Systeme sind ein Sicherheits- und Compliance-Risiko.

5 Netzwerk & Ports

Von	Nach	Port	Zweck
Client / Portal	API	443 (HTTPS)	REST-Zugriff
Browser	Portal	8080 (HTTPS)	Self-Service
API	PostgreSQL	5432	Datenbank
API	Mail-Server	587 (STARTTLS) / 993 (IMAPS)	Versand / Eingang

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Von	Nach	Port	Zweck
API	Zielsysteme	135/445 (WMI), 22 (SSH), 161 (SNMP), 389/636 (LDAP)	Discovery-Scan (serverseitig)
Discovery-Collector	Zielsysteme im Standortnetz	135/445 (WMI), 22 (SSH), 161 (SNMP), 389/636 (LDAP)	Discovery-Scan vor Ort
Discovery-Collector	API	443 (HTTPS)	Funde melden – die Verbindung geht **immer vom Standort aus**

> **Es gibt keinen eigenen Scan-Dienst auf dem Server.** Die serverseitigen Scans laufen im API-Dienst mit; die Verbindungen zu den Zielsystemen gehen also vom API-Host aus. Ein eigener Windows-Dienst existiert nur für entfernte Netze, die der Server nicht erreicht: der **Discovery-Collector** mit eigenem Installationspaket (siehe ID-50). Er nimmt keine eingehenden Verbindungen an.

5.1 Dünner Client – standortübergreifend, VPN, weltweit

Der Windows-Client ist ein **dünner Client**: Er spricht **ausschließlich** die REST-API an (HTTP/HTTPS) und verbindet sich **nie** direkt mit der Datenbank – er kann es technisch gar nicht (der Client enthält keinen Datenbanktreiber). Auch datenbanknahe Anzeigen holt er über die API.

Für den Betrieb folgt daraus:

- **Nur der API-Port** muss von den Arbeitsplätzen erreichbar sein. **PostgreSQL (5432) kann auf den Server bzw. localhost beschränkt bleiben** – die Datenbank ist aus dem Client-Netz gar nicht erreichbar (Sicherheitsgewinn).
- Der Client ist damit **standortübergreifend** einsetzbar: über **VPN**, zwischen mehreren Standorten, bei Bedarf **weltweit** – vorausgesetzt, die API wird **sicher veröffentlicht** (natives HTTPS/TLS, siehe Betriebshandbuch ID-50, Abschnitt „HTTPS einrichten") **oder** der Zugang läuft über ein **VPN**.
- Die eine API-Adresse (`http(s)://server:port`) wird im Client unter *Verbindungseinstellungen* bzw. bei der Installation gesetzt (Betriebshandbuch, Abschnitt 7.2).

> **Sicherheitshinweis:** Eine ins Internet veröffentlichte API immer mit HTTPS und starker Authentifizierung betreiben. Wo das nicht gewährleistet ist, den Zugang auf ein VPN beschränken.

6 Datensicherheit auf einen Blick

- **Verschlüsselung:** TLS für alle Verbindungen; Zugangsdaten für Discovery werden AES-256-verschlüsselt im Credential-Vault gespeichert.
- **Authentifizierung:** JWT-basiert; Berechtigungen werden als `permission` -Claims transportiert.
- **AD/LDAP-Anbindung:** immer **read-only** – Ordivis Platform schreibt niemals ins Active Directory.
- **Mandantentrennung:** serverseitige Query-Filter je Mandant.
- **Kein Direktzugriff auf die Datenbank:** Der Client spricht ausschließlich die API an; PostgreSQL ist aus dem Client-Netz nicht erreichbar, und der Client trägt **keine** Datenbank-Zugangsdaten (nichts, das am

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Arbeitsplatz gestohlen oder ausgelesen werden könnte). Jeder Datenzugriff wird API-seitig durch Authentifizierung, RBAC, Mandantentrennung, Validierung und Audit vermittelt – ein Vorbeischleusen an diesen Kontrollen, wie es direkter Datenbankzugriff erlauben würde, ist ausgeschlossen. Das entspricht den Grundsätzen der Netzsegmentierung und der minimalen Rechte (ISO 27001 / BSI-Grundschutz). Die Absicherung konzentriert sich damit auf die API: Diese ist folglich mit **HTTPS und starker Authentifizierung** zu betreiben bzw. der Zugang über **VPN** zu führen (siehe 5.1).

Details zu Härtung, Zertifikaten und Sicherung finden Sie im Betriebshandbuch (ID-50) sowie im ISMS-Modul (ID-28).

7 Editionen & Lizenzierung

Der Funktionsumfang kann durch die **Produktlizenz** begrenzt sein (z. B. Anzahl Administratoren oder Mandanten). Die Testphase umfasst 30 Tage mit vollem Funktionsumfang. Einzelheiten zur Aktivierung siehe ID-43.

Verwandte Themen

- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Installation, Update und Sicherung Schritt für Schritt
- **ID-51 – Datenbank-Leistung & Dimensionierung** – Dimensionierung und Verbindungspools nach Kundengröße
- **ID-42 – Modul Mandantenfähigkeit** – wie mehrere Organisationen in einer Installation getrennt werden
- **ID-28 – Modul Informationssicherheit (ISMS)** – Schutzbedarf und Härtung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL II · FÜR ENDBENUTZER

Benutzerhandbuch – Self-Service-Portal

Das Self-Service-Portal – einfach erklärt, Schritt für Schritt

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-10 Benutzer

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Benutzerhandbuch – Self-Service-Portal

Dieses Handbuch ist für **alle Mitarbeiterinnen und Mitarbeiter** gedacht, die über Ordivis Platform ein Anliegen melden möchten. Sie brauchen **keine IT-Vorkenntnisse**. Jeder Schritt ist genau beschrieben. Sie arbeiten dabei im **Portal**, das Sie einfach im Browser öffnen.

> **Alles in diesem Dokument spielt im Portal.** Deshalb steht in den Anleitungsköpfen nur 🌐 – der Windows-Client ist kein Werkzeug für Endanwender und bietet die Bereiche *Meine Vorgänge* und *Mein Arbeitsplatz* gar nicht erst an. Wer beide Oberflächen benutzt, findet die Gegenüberstellung in ID-01, Abschnitt 3.3.

INHALT

1 Was ist das Portal?
1.1 Sich im Menü zurechtfinden
2 Anmelden
3 Ihr Kennwort selbst ändern
4 Ein Anliegen melden (Ticket erstellen)
5 Eine Leistung anfordern (Service-Katalog)
6 Status verfolgen und auf Rückfragen antworten
6.1 Auf eine Rückfrage antworten
7 In der Wissensdatenbank nach Lösungen suchen
8 Die Hilfe bewerten (Zufriedenheit / CSAT)
9 Termine mit dem Service-Desk
10 Meine Geräte und eine Störung daran melden
11 Datenschutz – Ihre Rechte wahrnehmen
12 Change-Kalender – geplante Wartungen im Voraus sehen
13 Benachrichtigungen selbst einstellen
14 Awareness & Bestätigungen
15 Genehmigungs-Vertretung (für Genehmiger)
16 Das Portal als App installieren (offline)
17 Wetter und Unwetterwarnungen
17.1 Die Warnkarte lesen
18 Abmelden
19 Häufige Fragen
Verwandte Themen

1 Was ist das Portal?

Das Portal ist Ihre persönliche Anlaufstelle bei Fragen und Problemen rund um die IT. Hier können Sie:

- ein **Anliegen melden** (z. B. „Drucker geht nicht“),
- den **Status** Ihrer Anliegen verfolgen,
- **Leistungen anfordern** (z. B. neue Software),

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- in der **Wissensdatenbank** nach Lösungen suchen,
- auf **Rückfragen** der IT antworten.

Nach dem Anmelden landen Sie auf Ihrer persönlichen **Übersicht**: offene Anliegen, Aufgaben und Schnellzugriffe auf einen Blick.

So sieht diese Übersicht aus (Abbildung 1).

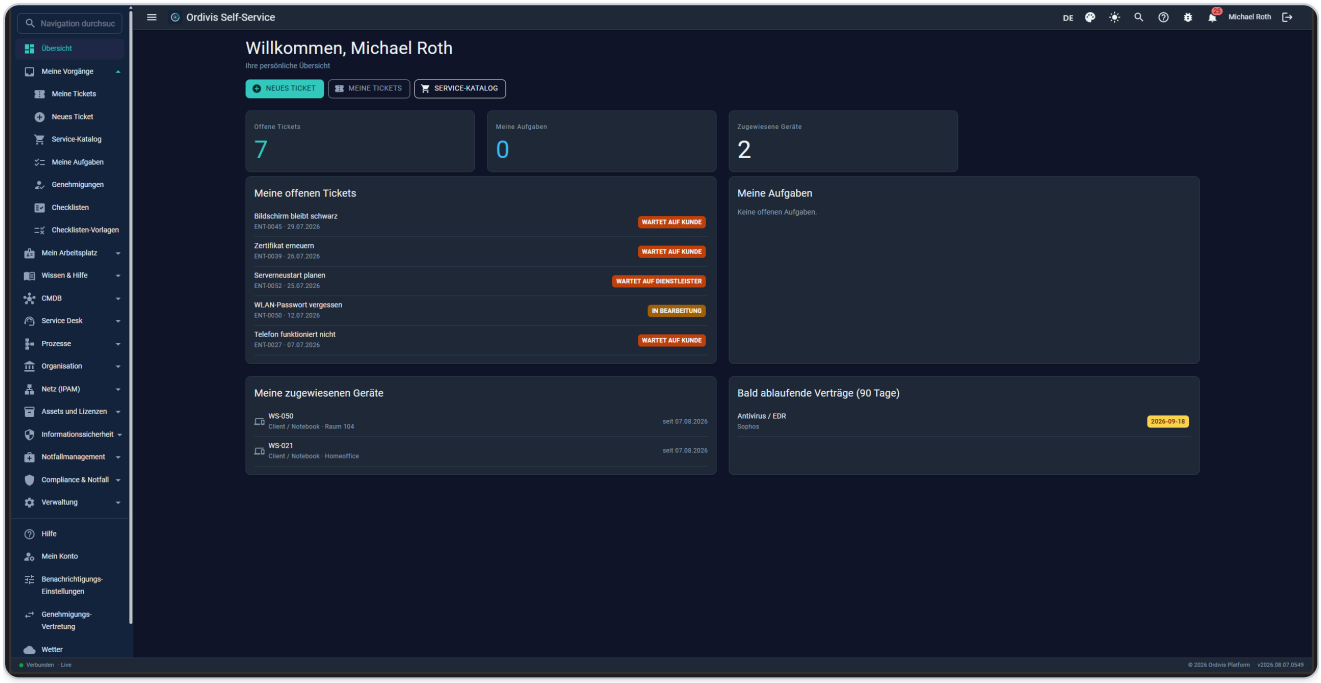

Abb. 1 – Ihre persönliche Übersicht: offene Tickets, Aufgaben und Schnellzugriffe (Neues Ticket, Meine Tickets, Service-Katalog).

Ein Wort vorweg: „Ticket". Auf dem Bildschirm heißt Ihr gemeldetes Anliegen *Ticket*. Das ist kein Fachchinesisch mit Hintersinn – es ist schlicht die Nummer, unter der Ihr Anliegen geführt wird, damit Sie und die IT über dieselbe Sache sprechen. Dieses Handbuch benutzt beide Wörter; gemeint ist dasselbe.

1.1 Sich im Menü zurechtfinden

Links steht das Menü, aufgeteilt in Bereiche. Für Sie zählen vor allem drei:

Bereich	Was Sie dort finden
Meine Vorgänge	*Meine Tickets*, *Neues Ticket*, *Service-Katalog*, *Meine Aufgaben*, *Genehmigungen*, *Checklisten*
Mein Arbeitsplatz	*Meine Geräte*, *Awareness & Bestätigungen*, *Datenschutz*
Wissen & Hilfe	*Wissensdatenbank*, *Bekannte Fehler (KEDB)*

Ganz unten, unterhalb der Bereiche, liegt der **Fußbereich** mit *Hilfe*, *Mein Konto*, *Benachrichtigungs-Einstellungen*, *Genehmigungs-Vertretung*, *Wetter*, *Systemstatus* und *Abmelden*.

> **Tipp:** Über dem Menü steht ein **Suchfeld**. Tippen Sie dort ein Stichwort ein, bleiben nur die passenden Menüeinträge stehen. Das ist schneller, als sich zu merken, in welchem Bereich etwas liegt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Hinweis: Sie sehen nur die Einträge, für die Sie berechtigt sind. Fehlt bei Ihnen ein hier beschriebener Punkt, ist das kein Fehler – wenden Sie sich an Ihre IT.

2 Anmelden

🌐 **Nur im Portal:** die Portal-Adresse im Browser · 🔑 kein Recht nötig

Voraussetzung: Sie haben von Ihrer IT die **Portal-Adresse**, einen **Benutzernamen** und ein **erstes Kennwort** erhalten. Ohne ein von der IT vergebenes Kennwort kommen Sie nicht hinein – ein neu angelegtes Konto hat zunächst keines.

1. Öffnen Sie Ihren Browser (z. B. Edge oder Chrome).
2. Geben Sie die **Portal-Adresse** in die Adresszeile ein (z. B. `https://portal.ihre-firma.de`) und drücken Sie die Eingabetaste.
3. Tippen Sie Ihren **Benutzernamen** in das erste Feld. Groß- oder Kleinschreibung spielt hier keine Rolle.
4. Tippen Sie Ihr **Kennwort** in das zweite Feld. Hier ist Groß-/Kleinschreibung wichtig.
5. Klicken Sie auf *Anmelden*.

✅ **Ergebnis:** Sie landen auf der Übersicht (Abbildung 1). Oben rechts steht Ihr Name.

Abbildung 2 zeigt die Anmeldeseite.

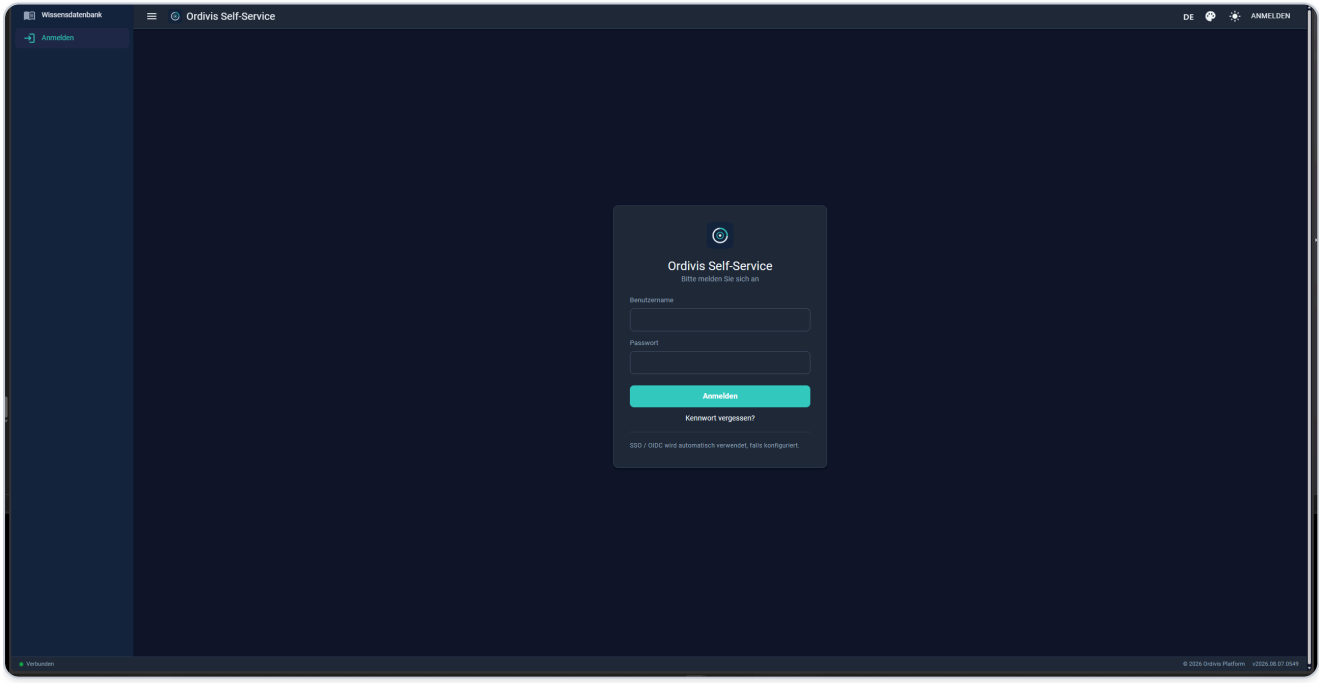

Abb. 2 – Die Anmeldeseite: Benutzername und Kennwort eingeben, dann „Anmelden“.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Hinweis: Ihr **erstes (initiales) Kennwort** wird von der IT/einem Administrator vergeben. Danach können Sie Ihr Kennwort **selbst ändern** (Abschnitt 3). Voraussetzung ist, dass Sie sich **mindestens einmal erfolgreich angemeldet** haben.

Tipp: Sie kennen Ihr Kennwort nicht mehr oder haben sich noch nie angemeldet? Wenden Sie sich an Ihre IT – nur ein Administrator kann ein **neues initiales** Kennwort vergeben.

3 Ihr Kennwort selbst ändern

🌐 **Nur im Portal:** *Fußbereich* ▸ *Mein Konto* · 🔑 kein Recht nötig – jeder darf sein eigenes Kennwort ändern

Voraussetzung: Sie haben sich mindestens einmal erfolgreich angemeldet und kennen Ihr **aktuelles** Kennwort. Ohne das aktuelle Kennwort geht es nicht – auch nicht über Umwege.

🔒 **Pflichtreihenfolge** — das neue Kennwort wird erst geprüft, wenn das aktuelle stimmt. Wer oben nichts einträgt, bekommt beim Speichern die Meldung *Bitte aktuelles Kennwort eingeben.*

1. Klicken Sie unten im Menü auf **Mein Konto**.
2. Wechseln Sie auf den Abschnitt **Kennwort ändern**.
3. Tippen Sie Ihr bisheriges Kennwort in das Feld *Aktuelles Kennwort*.
4. Tippen Sie Ihr Wunschkennwort in das Feld *Neues Kennwort*.
5. Tippen Sie dasselbe Kennwort noch einmal in das Feld *Neues Kennwort bestätigen*. Stimmen die beiden nicht überein, meldet das Portal *Die Kennwörter stimmen nicht überein.*
6. Klicken Sie auf *Kennwort ändern*.

✅ **Ergebnis:** Es erscheint die Meldung *Kennwort erfolgreich geändert.* Ab dem nächsten Anmelden gilt das neue Kennwort.

Tipp: Wählen Sie ein langes, einprägsames Kennwort (z. B. einen Satz). Verwenden Sie es nicht für andere Dienste.

Hinweis: Wird Ihr Konto aus dem Active Directory verwaltet, steht unter *Kontotyp* **Extern verwaltet**. Dann ändern Sie Ihr Kennwort dort, wo Sie sich auch am PC anmelden – nicht hier.

Hinweis: Ein Zurücksetzen bei **vergessenem** Kennwort über das Portal (ohne Administrator) ist geplant und wird in einer späteren Version ergänzt. Bis dahin wenden Sie sich bei einem vergessenen Kennwort an Ihre IT.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4 Ein Anliegen melden (Ticket erstellen)

> 🌐 **Nur im Portal:** *Meine Vorgänge* ▸ *Neues Ticket* · 🔑 kein Recht nötig – jeder darf ein eigenes Anliegen melden

Auf der Seite *Neues Ticket erstellen* haben Sie oben **zwei Schaltflächen** zur Wahl:

- *Service-Katalog* – für vorbereitete Leistungen, die Ihr Haus anbietet (Abschnitt 5).
- *Freie Anfrage* – für alles andere. Das ist der Weg in diesem Abschnitt.

🔒 **Pflichtreihenfolge** — die Artikelvorschläge (Schritt 4) entstehen aus dem, was Sie in *Titel* und *Beschreibung* geschrieben haben. Wer sie vor dem Schreiben sucht, sieht nichts.

1. Klicken Sie im Menü auf **Neues Ticket**.
2. Klicken Sie oben auf *Freie Anfrage*.
3. Schreiben Sie in das Feld *Titel* eine Zeile, die das Problem benennt (z. B. „Drucker im 2. OG druckt nicht“).
4. Beschreiben Sie im Feld *Beschreibung* möglichst genau: - Was funktioniert nicht? - Seit wann? - Welche Fehlermeldung erscheint (falls vorhanden)?
5. **Sehen Sie unter Ihrer Eingabe Artikelvorschläge?** Das Portal durchsucht während des Schreibens die Wissensdatenbank und blendet Passendes unter der Überschrift *Vielleicht hilft dir das schon weiter?* ein. Löst ein Artikel Ihr Problem, klicken Sie auf *Hat geholfen – kein Ticket nötig* – dann sind Sie fertig, und niemand muss sich damit befassen.
6. Wählen Sie unter *Kategorie*, worum es geht: *Störung (Incident)*, *Service Request* oder *Frage / Information*. Sind Sie unsicher, lassen Sie *Störung (Incident)* stehen.
7. Schätzen Sie unter *Priorität (Selbsteinschätzung)* die Dringlichkeit ein: *Niedrig*, *Mittel* oder *Hoch*.
8. Ist ein bestimmtes Gerät betroffen, wählen Sie es unter *Betroffenes Gerät (optional)* aus. Zur Auswahl stehen die Geräte, die auf Sie eingetragen sind.
9. Möchten Sie ein **Bild oder eine Datei** anhängen (z. B. einen Screenshot der Fehlermeldung), klicken Sie auf *Dateien anhängen* und wählen die Datei aus. Bis zu **zehn** Dateien sind möglich.
10. Klicken Sie auf *Ticket erstellen*.

✅ **Ergebnis:** Es erscheint *Ticket erfolgreich erstellt.* und Sie landen im neuen Vorgang. Er trägt eine **Ticketnummer** – damit finden Sie Ihr Anliegen jederzeit wieder.

Das Formular sehen Sie in Abbildung 3.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

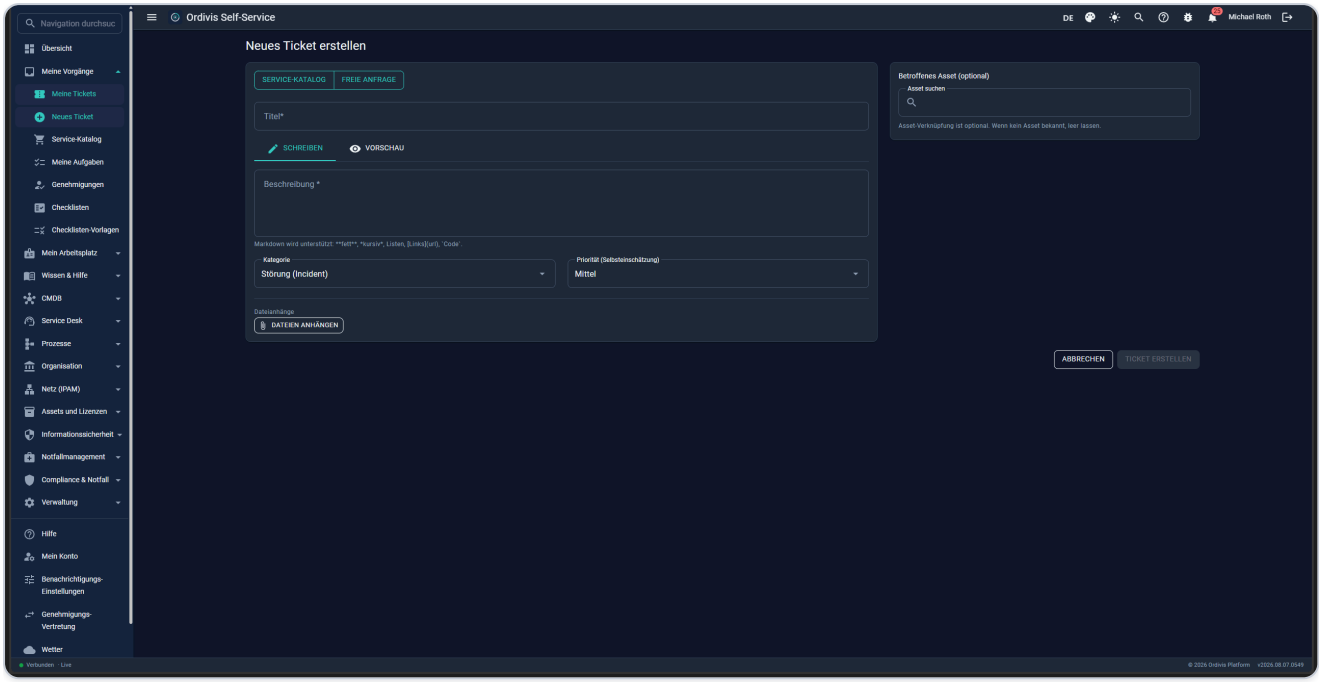

Abb. 3 – Neues Anliegen: vorbereitete Katalog-Anfragen oder eine freie Anfrage; optional ein betroffenes Gerät verknüpfen.

> **Tipp:** Je genauer Ihre Beschreibung, desto schneller kann die IT helfen. Ein Screenshot sagt oft mehr als viele Worte.

> **Hinweis:** Ihre *Priorität (Selbsteinschätzung)* ist eine Angabe für die IT, keine Festlegung. Die verbindliche Priorität ergibt sich aus Dringlichkeit und Auswirkung und wird im Service Desk gesetzt (ID-20).

5 Eine Leistung anfordern (Service-Katalog)

> 🌐 **Nur im Portal:** *Meine Vorgänge* ▸ *Service-Katalog* · 🔑 kein Recht nötig

Manche Dinge können Sie direkt „bestellen“, z. B. eine Software oder einen Zugang. Dafür gibt es den Katalog: Jede Leistung darin ist vorbereitet, mit fester Kategorie und bekannter Bearbeitungszeit.

🔒 **Pflichtreihenfolge** — der Assistent führt in drei Schritten (*Leistung → Angaben zur Anfrage → Prüfen & Absenden*). Vorwärts geht es nur über *Weiter*; zurück jederzeit über *Zurück*, ohne dass Eingaben verloren gehen.

1. Klicken Sie im Menü auf **Service-Katalog**.
2. Suchen Sie die gewünschte Leistung in der Liste. Unter jeder Kachel steht die **voraussichtliche Bearbeitungszeit**; ist eine Genehmigung nötig, trägt sie zusätzlich die Marke *Freigabepflichtig*.
3. Klicken Sie auf der Kachel auf *Anfragen*.
4. Füllen Sie unter *Angaben zur Anfrage* die abgefragten Felder aus. Welche das sind, hängt von der Leistung ab; Pflichtfelder sind gekennzeichnet.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5. Wählen Sie bei Bedarf unter *Betroffenes Gerät / CI (optional)* das Gerät aus, um das es geht.
6. Ergänzen Sie unter *Zusätzliche Anmerkungen*, was sonst noch wichtig ist.
7. Klicken Sie auf *Weiter*.
8. Prüfen Sie auf der Seite *Prüfen & Absenden* Ihre Angaben. Dort steht auch, ob eine **Freigabe** *erforderlich (Genehmigung durch Vorgesetzte/n)* ist oder *nicht erforderlich*.
9. Klicken Sie auf *Absenden*.

✅ **Ergebnis:** Es erscheint *Ihre Anfrage wurde als Ticket angelegt.* Ihre Anforderung läuft ab jetzt als ganz normaler Vorgang unter *Meine Tickets*.

Abbildung 4 zeigt den Katalog mit Bearbeitungszeit und Hinweis auf eine Freigabepflicht.

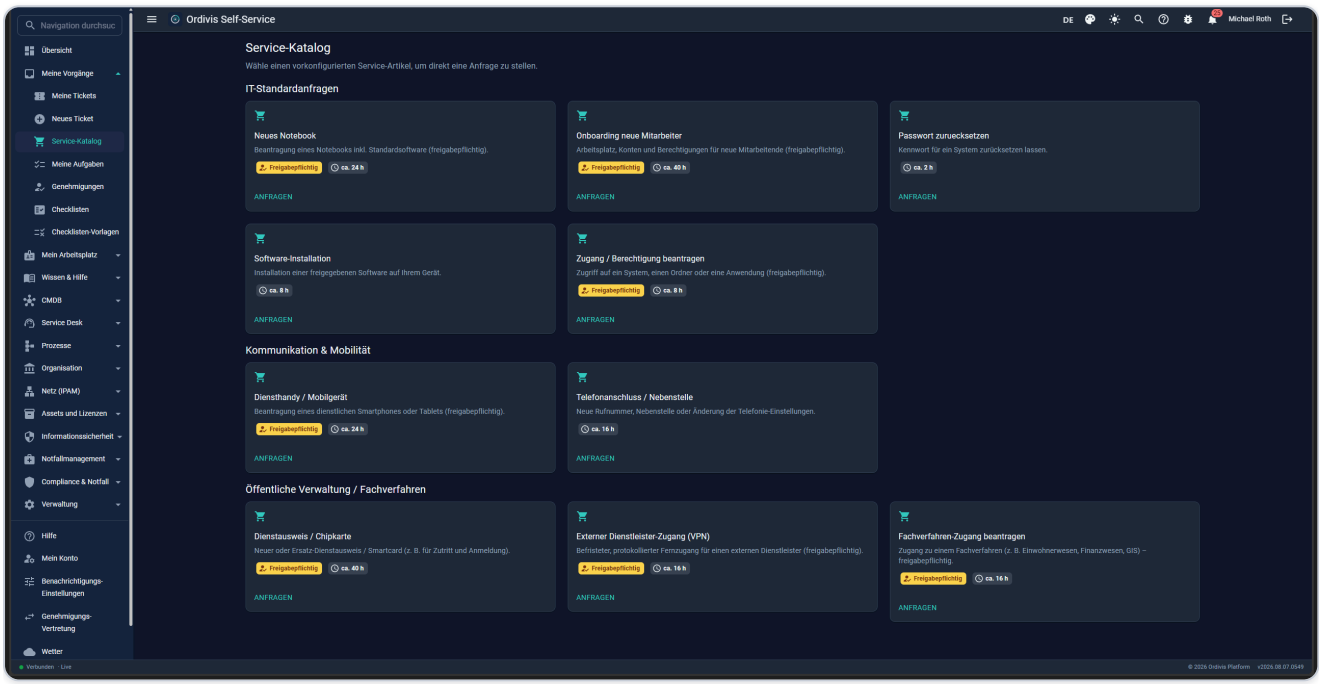

Abb. 4 – Der Servicekatalog: vorkonfigurierte Leistungen mit Bearbeitungszeit und Hinweis auf Freigabepflicht.

Wichtig: Bei einer freigabepflichtigen Leistung passiert nach dem Absenden zunächst **nichts sichtbar** – der Vorgang wartet auf die Entscheidung Ihrer Führungskraft. Das ist kein Stillstand, sondern der vorgesehene Weg. Den Stand sehen Sie im Ticket unter *Freigabe:*.

6 Status verfolgen und auf Rückfragen antworten

🌐 **Nur im Portal:** *Meine Vorgänge* ▸ *Meine Tickets* · 🔑 kein Recht nötig – Sie sehen ausschließlich Ihre eigenen Vorgänge

1. Klicken Sie im Menü auf **Meine Tickets**.
2. Sie sehen eine Tabelle mit *Nummer*, *Titel*, *Status*, *Priorität*, *Erstellt* und *Fällig am*. Der Status ist farbig hinterlegt (z. B. *Offen*, *In Bearbeitung*, *Gelöst*).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3. Suchen Sie einen bestimmten Vorgang, tippen Sie ein Stichwort in das Feld *Suchen*. Über die Auswahl daneben schalten Sie zwischen *Alles außer geschlossen* und *Alle* um – geschlossene Vorgänge sind sonst ausgeblendet.
4. Klicken Sie auf eine Zeile, um den Vorgang zu öffnen.

Wie diese Liste aussieht, zeigt Abbildung 5.

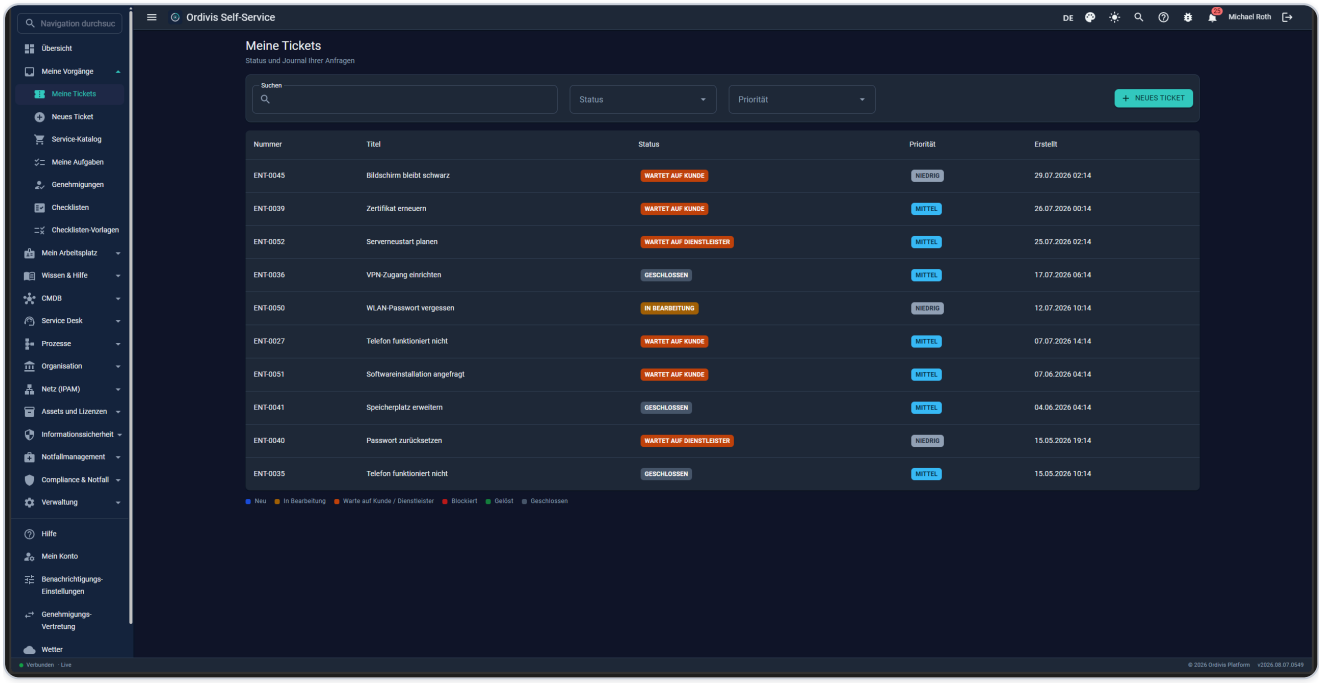

Abb. 5 – „Meine Tickets“: alle eigenen Anliegen mit Nummer, Titel und farbigem Status; such- und filterbar.

6.1 Auf eine Rückfrage antworten

> 🌐 **Nur im Portal:** *Meine Vorgänge* ▸ *Meine Tickets* ▸ *ein Ticket öffnen* · 🔑 kein Recht nötig

1. Öffnen Sie den Vorgang aus der Liste.
2. Lesen Sie im Bereich *Journal* den bisherigen Verlauf – er zeigt in zeitlicher Folge, wer wann was geschrieben hat.
3. Klicken Sie auf *Antworten*.
4. Schreiben Sie Ihre Antwort in das Textfeld.
5. Klicken Sie auf *Senden*.

✅ **Ergebnis:** Es erscheint *Antwort gesendet.*, und Ihr Beitrag steht als neuer Eintrag oben im *Journal*. Die IT wird benachrichtigt.

> **Hinweis:** Sie können auch **per E-Mail** antworten. Antworten Sie einfach auf die Benachrichtigungs-E-Mail – Ihre Antwort landet automatisch im richtigen Anliegen. Bitte den Betreff (mit der Ticketnummer) nicht verändern.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Hinweis: Einträge mit der Marke *Intern* sehen Sie nicht. Das sind Vermerke der Bearbeiter untereinander; sie werden Ihnen gar nicht erst übermittelt.

7 In der Wissensdatenbank nach Lösungen suchen

🌐 **Nur im Portal:** *Wissen & Hilfe* ▸ *Wissensdatenbank* · 🔑 kein Recht nötig für veröffentlichte Artikel

Oft finden Sie die Lösung selbst – schneller, als auf Hilfe zu warten.

1. Klicken Sie im Menü auf **Wissensdatenbank**.
2. Tippen Sie ein Stichwort in das Feld *Suchen* (z. B. „VPN einrichten“).
3. Grenzen Sie das Ergebnis bei Bedarf über *Kategorien* oder *Schlagworte* links weiter ein.
4. Klicken Sie auf einen Artikel, um ihn zu lesen.
5. Folgen Sie den beschriebenen Schritten.

✅ **Ergebnis:** Hilft der Artikel, ist Ihr Anliegen erledigt, ohne dass ein Ticket entsteht.

Tipp: Führt die Suche ins Leere, klicken Sie auf *zurücksetzen* und versuchen es mit einem **einzelnen** Wort. Zu viele Wörter grenzen zu stark ein – „VPN“ findet mehr als „VPN funktioniert nicht von zu Hause“.

8 Die Hilfe bewerten (Zufriedenheit / CSAT)

🌐 **Nur im Portal:** *Meine Vorgänge* ▸ *Meine Tickets* ▸ *ein gelöstes Ticket öffnen* · 🔑 kein Recht nötig

Wurde Ihr Anliegen **gelöst oder geschlossen**, können Sie die Bearbeitung bewerten. Ihre Rückmeldung hilft der IT, besser zu werden.

1. Öffnen Sie den gelösten Vorgang.
2. Klicken Sie rechts im Abschnitt *Wie zufrieden waren Sie mit der Lösung?* auf die Anzahl Sterne (1 bis 5), die Ihre Erfahrung am besten beschreibt.
3. Ergänzen Sie bei Bedarf einen Text unter *Kommentar (optional)*.
4. Klicken Sie auf *Bewertung abgeben*. Ohne mindestens einen Stern bleibt die Schaltfläche deaktiviert.

✅ **Ergebnis:** Es erscheint *Danke für Ihre Bewertung!*

Wichtig: Bewerten lässt sich **einmalig** je Vorgang. Danach ist die Schaltfläche weg – eine Korrektur ist nicht vorgesehen. Nehmen Sie sich die zehn Sekunden für den Kommentar; die Zahl allein sagt der IT nicht, was gut oder schlecht war.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

9 Termine mit dem Service-Desk

🌐 **Nur im Portal:** *Meine Vorgänge* ▸ *Meine Tickets* ▸ *ein Ticket öffnen* · 🔑 kein Recht nötig

Schlägt Ihnen die IT im Ticket einen Termin vor, entscheiden Sie darüber selbst.

1. Öffnen Sie den Vorgang.
2. Suchen Sie den Abschnitt *Terminvorschläge* mit Datum, Uhrzeit und – falls angegeben – Ort.
3. Klicken Sie auf *Zusagen*, wenn der Termin passt, sonst auf *Absagen*.
4. Möchten Sie den Termin im eigenen Kalender haben, klicken Sie auf den Link *.ics* daneben und öffnen Sie die heruntergeladene Datei. Outlook und vergleichbare Programme tragen den Termin dann selbst ein.

✅ **Ergebnis:** Es erscheint *Termin zugesagt.* bzw. *Termin abgesagt.*, und Ihre Antwort wird im *Journal* des Vorgangs vermerkt – die IT sieht sie sofort.

Hinweis: Sie bekommen den Vorschlag zusätzlich per E-Mail. Darin stecken dieselben zwei Links; ein Klick genügt, ohne Anmeldung im Portal.

10 Meine Geräte und eine Störung daran melden

🌐 **Nur im Portal:** *Mein Arbeitsplatz* ▸ *Meine Geräte* · 🔑 kein Recht nötig – Sie sehen ausschließlich die auf Sie eingetragenen Geräte

Die Seite listet Ihre Geräte mit *Gerät*, *Klasse*, *Seriennummer*, *Zugewiesen* und – sofern hinterlegt – *Garantie bis*. Steht dort *Ihnen sind aktuell keine Geräte zugewiesen.*, führt Ihre IT die Zuordnung (noch) nicht; melden Sie eine Störung dann über den Weg in Abschnitt 4.

So melden Sie ein defektes Gerät:

1. Klicken Sie im Menü auf **Meine Geräte**.
2. Suchen Sie das betroffene Gerät in der Liste.
3. Klicken Sie in seiner Zeile auf *Defekt melden*.
4. Beschreiben Sie im Feld *Was ist das Problem?*, was nicht funktioniert.
5. Klicken Sie auf *Störung anlegen*.

✅ **Ergebnis:** Es erscheint *Störung angelegt.* Der neue Vorgang ist **bereits mit dem Gerät verknüpft** – die IT sieht Modell und Seriennummer, ohne nachfragen zu müssen.

Tipp: Dieser Weg ist dem freien Ticket vorzuziehen, wann immer ein Gerät im Spiel ist. Die Rückfrage „Welches Gerät genau?“ entfällt, und das kostet sonst regelmäßig einen halben Tag.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

11 Datenschutz – Ihre Rechte wahrnehmen

🌐 **Nur im Portal:** *Mein Arbeitsplatz* ▸ *Datenschutz* · 🔑 kein Recht nötig – das Recht auf Auskunft hat jede betroffene Person

Auf der Seite *Datenschutz – Meine Rechte* stellen Sie Anträge zu den personenbezogenen Daten, die Ihr Haus über Sie führt.

1. Klicken Sie im Menü auf **Datenschutz**.
2. Wählen Sie unter *Art des Antrags* aus: - *Auskunft / Datenkopie (Art. 15 DSGVO)* – welche Daten liegen über mich vor? - *Berichtigung meiner Daten (Art. 16 DSGVO)* – etwas ist falsch gespeichert. - *Löschung meiner Daten (Art. 17 DSGVO)* – Daten sollen entfernt werden.
3. Schreiben Sie unter *Erläuterung (optional)*, worum es konkret geht. Bei einer Berichtigung gehört hierhin, **was** falsch ist und **wie** es richtig lautet.
4. Klicken Sie auf *Antrag absenden*.

✅ **Ergebnis:** Der Antrag geht an die Datenschutzbeauftragte oder den Datenschutzbeauftragten Ihres Hauses und wird dort bearbeitet (ID-32).

Hinweis: Die gesetzliche Frist läuft ab dem Eingang des Antrags. Ein früher gestellter Antrag wird also nicht schneller beantwortet, weil Sie ihn zusätzlich per E-Mail schicken – stellen Sie ihn hier, dann ist er nachweisbar erfasst.

12 Change-Kalender – geplante Wartungen im Voraus sehen

🌐 **Nur im Portal:** *Service Desk* ▸ *Change-Kalender* · 🔑 `change.read`

1. Klicken Sie im Menü auf **Change-Kalender**.
2. Lesen Sie die geplanten Änderungen in zeitlicher Reihenfolge, je mit *Nummer*, *Titel*, *Typ*, *Status* und *Risiko*.

✅ **Ergebnis:** Sie wissen im Voraus, wann mit Wartungen und Ausfällen zu rechnen ist.

Hinweis: Der Kalender verlangt das Recht `change.read`. Fehlt der Menüpunkt bei Ihnen, ist das keine Störung – Ihre Rolle sieht ihn nicht vor.

13 Benachrichtigungen selbst einstellen

🌐 **Nur im Portal:** *Fußbereich* ▸ *Benachrichtigungs-Einstellungen* · 🔑 kein Recht nötig

Sie legen je Ereignis fest, ob Sie *Im Portal* und/oder per *E-Mail* informiert werden möchten.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Klicken Sie unten im Menü auf **Benachrichtigungs-Einstellungen**.
2. Suchen Sie in der Spalte *Ereignis* die Zeile, um die es Ihnen geht.
3. Setzen oder entfernen Sie das Häkchen in der Spalte *Im Portal*.
4. Setzen oder entfernen Sie das Häkchen in der Spalte *E-Mail*.

✅ **Ergebnis:** Es erscheint *Gespeichert.* – die Änderung wirkt sofort. Einen Speichern-Knopf gibt es hier bewusst nicht.

Es gibt dafür genau **eine** Stelle. Unter *Mein Konto* steht nur der Weg dorthin (*Benachrichtigungs-Einstellungen öffnen*).

> **Nicht zu verwechseln mit der Glocke.** Das Symbol in der Kopfleiste führt zu den *erhaltenen* Meldungen — Ihrem Posteingang. Die Einstellungen bestimmen, welche davon überhaupt entstehen. Beides trug früher einen ähnlichen Namen und pflegte teils dieselben Daten; nach der Zusammenführung ist die Zuständigkeit eindeutig.

14 Awareness & Bestätigungen

> 🌐 **Nur im Portal:** *Mein Arbeitsplatz* ▸ *Awareness & Bestätigungen* · 🔑 kein Recht nötig

Weist Ihnen die IT eine **Richtlinie** oder eine **ISMS-Maßnahme** zu, erscheint sie hier – mit einem Datum unter *Fällig bis*.

1. Klicken Sie im Menü auf **Awareness & Bestätigungen**.
2. Öffnen Sie den Eintrag und lesen Sie den Text vollständig.
3. Klicken Sie auf *Gelesen & bestätigen*.

✅ **Ergebnis:** Es erscheint *Bestätigt.*, und der Eintrag wandert in den erledigten Teil der Liste. Steht dort *Keine offenen Bestätigungen – alles erledigt.*, haben Sie nichts zu tun.

> **Wichtig:** Ihre Bestätigung ist ein **Nachweis** und wird mit Zeitstempel gespeichert. Sie ist in einem Audit die Antwort auf die Frage, ob die Belegschaft die Richtlinie erhalten hat. Bestätigen Sie deshalb erst, wenn Sie den Text wirklich gelesen haben.

15 Genehmigungs-Vertretung (für Genehmiger)

> 🌐 **Nur im Portal:** *Fußbereich* ▸ *Genehmigungs-Vertretung* · 🔑 `change.approve`, `contracts.manage` oder `workflow.manage`

Wenn Sie Freigaben erteilen und **abwesend** sind, übertragen Sie Ihre Freigaben für einen Zeitraum an eine Vertretung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

🔒 **Pflichtreihenfolge** — der Zeitraum wird beim Anlegen festgeschrieben. Ändern lässt er sich danach nicht; es bleibt nur *Widerrufen* und neu anlegen.

1. Klicken Sie unten im Menü auf **Genehmigungs-Vertretung**.
2. Klicken Sie auf *Neue Vertretung*.
3. Wählen Sie unter *Vertretung* die Person aus, die für Sie einspringt.
4. Setzen Sie *Von* auf den ersten Tag Ihrer Abwesenheit.
5. Setzen Sie *Bis* auf den letzten Tag.
6. Halten Sie unter *Notiz (optional)* den Grund fest (z. B. „Urlaub“). Das erspart später die Rückfrage, warum jemand anders entschieden hat.
7. Klicken Sie auf *Delegation anlegen*.

✅ **Ergebnis:** Es erscheint *Delegation angelegt.* Der Eintrag steht unter *Meine Delegationen* und trägt im Zeitraum die Marke *Aktiv*.

So beenden Sie eine Vertretung vorzeitig: Klicken Sie in der Zeile auf *Widerrufen*. Es erscheint *Widerrufen.*, und ab sofort entscheiden Sie wieder selbst.

> **Hinweis:** Unter *Sie vertreten aktuell:* steht, für wen **Sie** gerade einspringen. Prüfen Sie das nach jedem Urlaub eines Kollegen – eine vergessene Vertretung erklärt so manche Freigabe, die man sich nicht erklären kann.

16 Das Portal als App installieren (offline)

> 🌐 **Nur im Portal:** Funktion Ihres Browsers · 🔑 kein Recht nötig

Das Portal lässt sich als **App** installieren (Progressive Web App). Es bekommt dann ein eigenes Symbol und ein eigenes Fenster ohne Adresszeile.

1. Öffnen Sie das Portal im Browser und melden Sie sich an.
2. Klicken Sie in der Adresszeile auf das **Installieren-Symbol** – in Edge und Chrome ein kleiner Bildschirm mit Pfeil am rechten Rand.
3. Bestätigen Sie mit *Installieren*. Auf dem Mobiltelefon heißt der Punkt im Browsermenü stattdessen *Zum Startbildschirm hinzufügen*.

✅ **Ergebnis:** Das Portal startet künftig über ein eigenes Symbol. Einmal geöffnete **Notfalldokumente** stehen danach auch **ohne Internetverbindung** zur Verfügung.

> **Wichtig:** Offline stehen nur Dokumente bereit, die Sie **vorher schon einmal geöffnet** haben. Der Notfall ist der falsche Zeitpunkt, das zu bemerken – öffnen Sie die für Sie wichtigen Notfalldokumente einmal im Normalbetrieb.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

17 Wetter und Unwetterwarnungen

🌐 **Nur im Portal:** *Fußbereich* ▸ *Wetter* · 🔑 kein Recht nötig – eine Unwetterwarnung betrifft jeden im Haus, nicht nur eine Fachrolle

Die Seite zeigt fünf Dinge, von oben nach unten:

1. **Amtliche Warnungen** des Deutschen Wetterdienstes ganz oben – farbig nach Stufe, mit Zeitraum und den Handlungsempfehlungen im Wortlaut des DWD. Sie stehen bewusst an erster Stelle: Bei Sturm, Starkregen oder Gewitter ist das die Angabe, auf die es ankommt.
2. Die **amtliche Warnkarte** (Abschnitt 17.1).
3. Die **aktuelle Lage** für Ihren Standort – Temperatur, Wetterlage, Luftfeuchte, mit dem Zeitpunkt der Messung.
4. Die **Brandgefahr** – Waldbrand- und Graslandfeuerindex, jeweils Stufe 1 (sehr gering) bis 5 (sehr hoch).
5. Eine **Vorschau auf die nächsten drei Tage** mit Höchst- und Tiefstwert, Niederschlag, Wind und Sonnenstunden.

Ist die Seite leer und meldet, der Wetterdienst sei nicht erreichbar, fehlt Ihrem Server der Zugang zum Internet. Wenden Sie sich in dem Fall an Ihre IT – alle übrigen Portalfunktionen sind davon nicht betroffen.

Welche Warnstufen es gibt. Vorinformation → Wetterwarnung → markante Wetterwarnung → Unwetterwarnung → extremes Unwetter. Die Angaben stammen unverändert vom Deutschen Wetterdienst; Ordivis Platform gibt sie nur weiter.

Die Warnungen gelten für Ihre Gemeinde, nicht für die nächste Wetterstation. Der DWD warnt für sogenannte Warnzellen – in aller Regel Gemeinden. Der Ort ergibt sich aus der **Postleitzahl im Mandantenprofil**. Fehlt sie dort, sagt die Seite das oben ausdrücklich und zeigt einen Standardort; die Warnungen gelten dann **nicht** für Ihre Gegend. Melden Sie das Ihrer IT.

17.1 Die Warnkarte lesen

Die Karte stammt unverändert vom Deutschen Wetterdienst. Darüber stellen Sie ein, was sie zeigt:

Schalter	Wirkung
Ihre Gegend / Ganz Deutschland	Ausschnitt um Ihren Ort oder das ganze Bundesgebiet
Warnart	Nur *eine* Art anzeigen: Alle, UV, Hitze, Gewitter, Wind, Regen, Schnee, Glatteis/Frost, Tauwetter, Nebel
Regenradar	Aktuellen Niederschlag über die Warnflächen legen
Blitze und Gewitterzellen	Zugbahnen und Blitze der letzten Minuten, mit Umkreis 5 und 10 km
Kleiner / Größer	Bildhöhe ändern – für den großen Schirm oder das schmale Fenster

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

So stellen Sie die Karte auf ein heranziehendes Gewitter ein:

> 🌐 **Nur im Portal:** *Fußbereich* ▸ *Wetter*, Schalter über der Karte · 🔑 kein Recht nötig

💡 **Empfohlene Reihenfolge** — zuerst die Warnart, dann die Blitze. Umgekehrt geht es auch, dann suchen Sie die Blitze aber eine Weile unter der flächigen Einfärbung.

1. Stellen Sie den Ausschnitt auf *Ihre Gegend*.
2. Wählen Sie unter *Warnart* den Eintrag *Gewitter*.
3. Schalten Sie *Blitze und Gewitterzellen* ein.
4. Schalten Sie bei Bedarf zusätzlich *Regenradar* ein, um die Niederschlagsfront zu sehen.

✅ **Ergebnis:** Die Karte zeigt nur noch Gewitterwarnungen, darüber die Zugbahnen und Blitze der letzten Minuten mit Umkreis 5 und 10 km.

> **Wozu die Auswahl der Warnart gut ist.** Eine bundesweite Hitzewarnung färbt die Karte flächendeckend ein. Darunter ist von einem heranziehenden Gewitter nichts mehr zu sehen – und genau dafür ist diese Seite da. Stellen Sie in so einer Lage **Gewitter** ein; die Blitze bleiben davon unberührt und lassen sich dazuschalten.

Unter der Karte steht die **Zeichenerklärung** des DWD mit der Bedeutung der Farben.

> **Die Waldbrandwarnstufe spricht die Forstbehörde des Landes aus**, nicht der DWD. Der hier gezeigte Index ist die meteorologische Grundlage dafür – nicht die amtliche Stufe selbst.

18 Abmelden

> 🌐 **Nur im Portal:** *Fußbereich* ▸ *Abmelden* · 🔑 kein Recht nötig

1. Klicken Sie unten im Menü auf *Abmelden*.

✅ **Ergebnis:** Sie landen wieder auf der Anmeldeseite. Ihre Sitzung ist beendet.

> **Achtung:** An **gemeinsam genutzten Computern** ist das Abmelden Pflicht. Das bloße Schließen des Browserfensters beendet die Sitzung nicht sofort – der oder die Nächste käme ohne Kennwort an Ihre Vorgänge.

> **Hinweis:** Ohne Zutun endet Ihre Sitzung nach **30 Minuten** von selbst. Sie werden dann beim nächsten Klick zur Anmeldung geschickt; ungespeicherte Eingaben gehen dabei verloren.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

19 Häufige Fragen

Ich habe die Ticketnummer vergessen. Öffnen Sie *Meine Tickets* – dort sind alle Ihre Vorgänge aufgelistet. Steht der gesuchte nicht dabei, stellen Sie die Auswahl oben von *Alles außer geschlossen* auf *Alle* um.

Mein Anliegen ist dringend. Beschreiben Sie die Dringlichkeit im Text und setzen Sie *Priorität (Selbsteinschätzung)* auf *Hoch*. Die verbindliche Priorität bewertet die IT. Bei einem kompletten Ausfall (z. B. „nichts geht mehr“) melden Sie sich zusätzlich telefonisch, falls Ihre IT eine Notfallnummer bereitstellt.

Ich sehe die Portal-Seite nicht. Prüfen Sie die Adresse und Ihre Internetverbindung. Hilft das nicht, wenden Sie sich an Ihre IT.

Ein Menüpunkt aus diesem Handbuch fehlt bei mir. Das Menü zeigt nur, wofür Sie berechtigt sind und welche Module Ihr Haus eingeschaltet hat. Fragen Sie Ihre IT, ob Ihre Rolle den Punkt vorsieht (ID-44).

Ich wurde plötzlich abgemeldet. Nach 30 Minuten ohne Zutun endet die Sitzung. Melden Sie sich neu an; Ihre Vorgänge sind unverändert da.

Verwandte Themen

- **ID-25 – Modul Wissensdatenbank** – Lösungsartikel selbst finden, bevor Sie ein Ticket eröffnen
- **ID-12 – Darstellung und Barrierefreiheit** – Darstellung und Barrierefreiheit auf Ihre Bedürfnisse einstellen
- **ID-34 – Modul Selbstorganisation** – eigene Aufgaben und Termine im Blick behalten
- **ID-32 – Modul Datenschutz (Art. 30 DSGVO)** – Ihre Rechte an den über Sie gespeicherten Daten
- **ID-01 – Einführung & Grundlagen** – Abschnitt 3.3 stellt Client und Portal gegenüber

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL III · FÜR AGENTEN

Agentenhandbuch – Service Desk

Tickets effizient bearbeiten – Liste, Detail, Kommunikation, Wissen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-11 Agenten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Agentenhandbuch – Service Desk

Dieses Handbuch richtet sich an **Service-Desk-Agenten**. Die vollständige Konfiguration (SLA, Kategorien, Routing) beschreibt das Modul-Dokument ID-20; die Wissensdatenbank ist in ID-25 vertieft.

> **Sie können in beiden Oberflächen arbeiten.** Seit Version 2026.08 ist der Agenten-Arbeitsplatz auch im Web-Portal vollständig da – mit Warteschlangen, Zuweisung, Statuswechsel, internen Notizen, Textbausteinen, Zeiterfassung und Eskalation. Jede Anleitung nennt deshalb beide Wege. Der Client bleibt der mächtigere Arbeitsplatz (gespeicherte Ansichten, Export, Zusammenführen, Aufteilen); das Portal ist der Weg vom Tablet, vom Mobiltelefon und aus dem Homeoffice.

Wo eine Funktion nur in einer Oberfläche existiert, sagt der Anleitungskopf das ausdrücklich.

INHALT

1 Die Ticket-Liste
- 1.1 Die Liste eingrenzen
- 1.2 Eine Ansicht speichern
- 1.3 Warteschlangen im Portal
- 1.4 Mehrere Vorgänge auf einmal ändern

2 Das Ticket-Detail
- 2.1 Mit dem Melder kommunizieren
- 2.2 Interne Notiz
- 2.3 Zuweisen, Status, Priorität und Zeit erfassen
- 2.4 Eskalieren
- 2.5 Termin mit dem Melder vereinbaren

3 Kategorie, Dringlichkeit, Auswirkung & Priorität
4 SLA-Fristen & Eskalation
5 ITIL-Prozesse im Alltag
6 Major Incident, Verknüpfen, Zusammenführen & Aufteilen
- 6.1 Ticket aufteilen

7 Wissen pflegen
8 Textbausteine (Canned Responses)
9 Einen Fehler an den Support melden
10 Tipps für effizientes Arbeiten
Verwandte Themen

1 Die Ticket-Liste

> 💻 **Client:** *Service Desk* ▸ *Tickets* · 🌐 **Portal:** *Service Desk* ▸ *Service-Desk-Vorgänge* · 🔑 `tickets.read`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Die Ticket-Liste ist das Herzstück Ihrer Arbeit. Abbildung 1 zeigt sie im Client mit Filtern, Export und gespeicherten Ansichten.

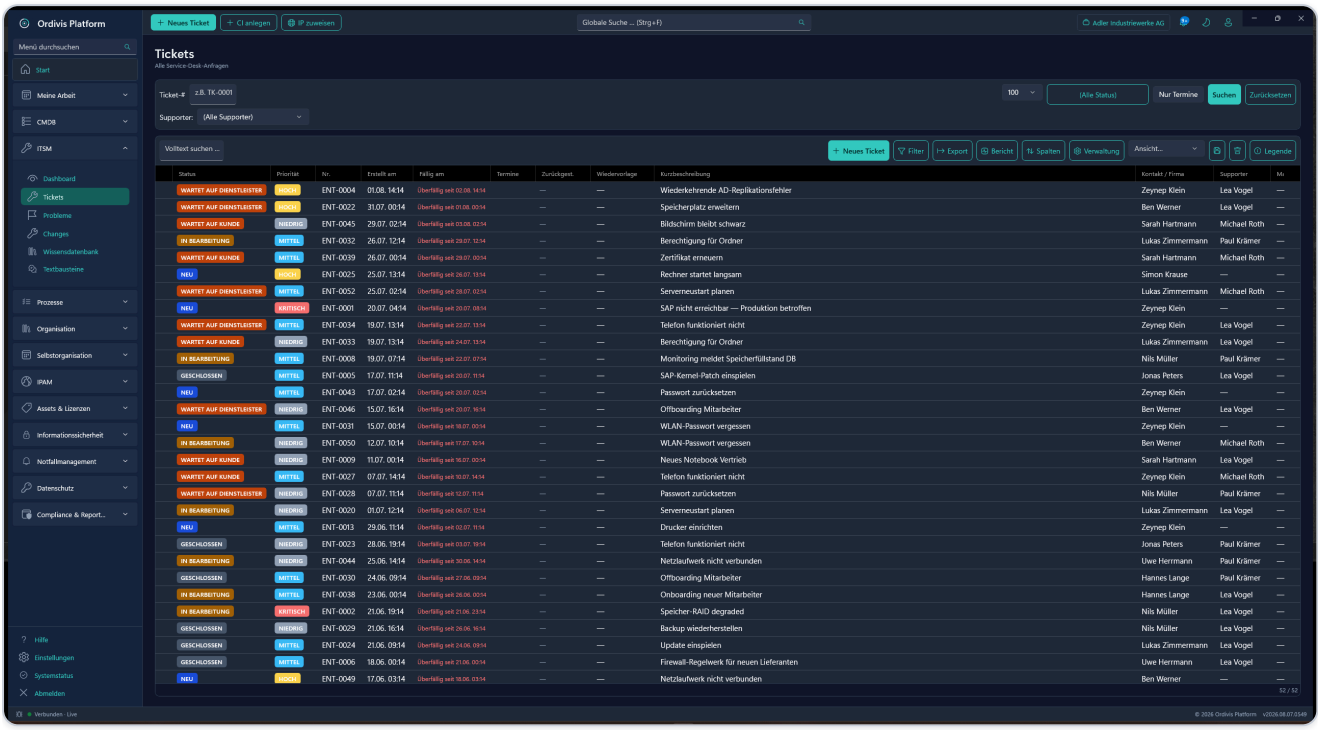

Abb. 1 – Die Ticket-Liste: farbige Status- und Prioritäts-Markierungen, Filter, Export und gespeicherte Ansichten.

> **Hinweis:** Agenten sehen standardmäßig nur Tickets ihrer **Support-Gruppe(n)**. Das Recht `tickets.read.all` hebt diese Einschränkung auf.

1.1 Die Liste eingrenzen

> 💻 **Client:** *Service Desk* ▸ *Tickets*, Werkzeugleiste über der Tabelle · 🔑 `tickets.read`

💡 **Empfohlene Reihenfolge** — erst filtern, dann suchen. Umgekehrt geht es auch, dann durchsucht die Volltextsuche aber den gesamten Bestand und Sie sortieren das Ergebnis von Hand nach.

1. Klicken Sie auf *Filter* und schränken Sie nach *Status*, *Priorität*, *Kategorie*, *Supporter* oder Support-Gruppe ein.
2. Tippen Sie bei Bedarf einen Begriff in das Feld *Volltext suchen …*. Die Suche greift **auch auf das Journal** zu, nicht nur auf den Titel – so finden Sie ein Ticket über einen Satz, an den Sie sich aus dem Verlauf erinnern.
3. Klicken Sie auf eine **Spaltenüberschrift**, um danach zu sortieren; ein zweiter Klick dreht die Richtung um.
4. Über *Spalten* blenden Sie Spalten ein und aus; über *Spaltenfilter* filtern Sie je Spalte.
5. Sind Sie fertig, setzen Sie mit *Zurücksetzen* alles auf den Ausgangszustand zurück.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

✅ **Ergebnis:** Die Tabelle zeigt genau die Vorgänge, um die es Ihnen geht. Filter, Sortierung und Spalten wirken zusammen und gelten auch für den Export.

1.2 Eine Ansicht speichern

> 💻 **Nur im Client:** *Service Desk* ▸ *Tickets* ▸ *Ansicht…* · 🔑 `tickets.read` — Das Portal bietet stattdessen feste Warteschlangen (Abschnitt 1.3).

🔒 **Pflichtreihenfolge** — gespeichert wird, was gerade eingestellt ist. Erst filtern, dann speichern; umgekehrt sichern Sie eine leere Ansicht.

1. Stellen Sie Filter, Sortierung und Spalten so ein, wie Sie sie täglich brauchen.
2. Klicken Sie auf *Ansicht…*.
3. Wählen Sie *Aktuelle Filter als Ansicht speichern*.
4. Geben Sie einen sprechenden Namen ein (z. B. „Meine offenen, nach Frist“) und bestätigen Sie.

✅ **Ergebnis:** Die Ansicht steht künftig unter *Ansicht…* ▸ *Gespeicherte Ansicht anwenden* und ist mit einem Klick wieder da. Über *Gewählte Ansicht löschen* entfernen Sie sie wieder.

1.3 Warteschlangen im Portal

> 🌐 **Nur im Portal:** *Service Desk* ▸ *Service-Desk-Vorgänge* · 🔑 `tickets.read` — Der Client arbeitet stattdessen mit frei gespeicherten Ansichten (Abschnitt 1.2).

Statt frei zusammengestellter Ansichten bietet das Portal oben eine Auswahl *Warteschlange* mit sechs festen Sichten:

Warteschlange	Zeigt
Offene Vorgänge	alles, was nicht geschlossen ist
Mir zugewiesen	Ihre eigenen Vorgänge
Nicht zugewiesen	was noch niemand übernommen hat
SLA verletzt	Vorgänge mit gerissener Frist
Wiedervorlage	zurückgestellte Vorgänge mit *Zurückgestellt bis*
Alle	ohne Einschränkung

1. Wählen Sie oben die gewünschte *Warteschlange*.
2. Klicken Sie auf eine Zeile, um den Vorgang zu öffnen.

✅ **Ergebnis:** Die Tabelle zeigt *Nummer*, *Titel*, *Typ*, *Status*, *Priorität*, *Melder*, *Bearbeiter*, *Erstellt* und *Fällig*. Steht dort *Keine Vorgänge in dieser Warteschlange*, ist die Sicht leer – das ist eine gute Nachricht, keine Störung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1.4 Mehrere Vorgänge auf einmal ändern

> 🌐 **Nur im Portal:** *Service Desk* ▸ *Service-Desk-Vorgänge* · 🔑 `tickets.update` — Im Client stehen dieselben Sammelaktionen als *Zuweisen*, *Zusammenführen* und *Schließen* in der Werkzeugleiste.

1. Setzen Sie in der Tabelle die Häkchen bei den Vorgängen, die Sie ändern möchten.
2. Klicken Sie oben auf *Zuweisen an…* und wählen Sie die Person – oder auf *Status setzen…* und wählen Sie den neuen Status.
3. Bestätigen Sie.

✅ **Ergebnis:** Es erscheint *{n} Vorgänge zugewiesen.* bzw. *{n} Vorgänge geändert.* Jede einzelne Änderung steht im Journal des jeweiligen Vorgangs – eine Sammelaktion ist kein Sonderweg an der Nachvollziehbarkeit vorbei.

> **Achtung:** Sammelaktionen fragen **nicht** einzeln nach. Prüfen Sie die Auswahl, bevor Sie bestätigen – im Client hilft dafür die Anzeige *n Ticket(s) ausgewählt* und die Schaltfläche *Auswahl aufheben*.

2 Das Ticket-Detail

> 🖥️ **Client:** *Service Desk* ▸ *Tickets*, Doppelklick auf eine Zeile · 🌐 **Portal:** *Service Desk* ▸ *Service-Desk-Vorgänge*, Klick auf eine Zeile · 🔑 `tickets.read`

Den Aufbau der Detailseite zeigt Abbildung 2.

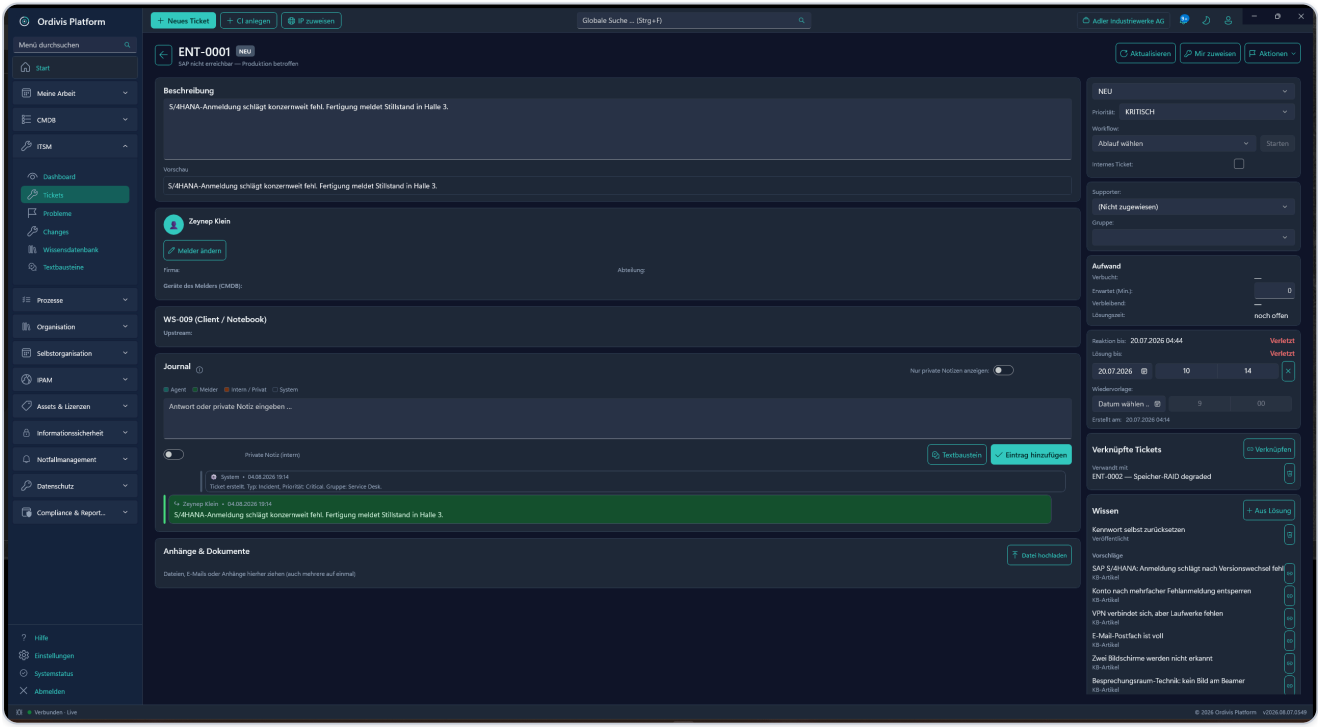

Abb. 2 – Ticket-Detail: Beschreibung, Journal (farbcodiert nach Quelle), verknüpftes CI und die Info-Leiste mit Status, Priorität, Aufwand und SLA-Frist.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Die Seite gliedert sich in vier Teile:

- **Kopfdaten:** Melder, Bearbeiter, Kategorie, Status, Priorität, SLA-Fristen.
- **Journal / Kommunikation:** der zeitliche Verlauf aus Nachrichten, Notizen und Systemeinträgen – dreistufig eingerückt für gute Lesbarkeit.
- **Private Notizen:** nur intern sichtbar, nicht für den Melder.
- **Anhänge:** Dateien zum Vorgang.

2.1 Mit dem Melder kommunizieren

> 💻 **Client:** Ticket-Detail ▸ *Journal* · 🌐 **Portal:** Ticket-Detail ▸ *Journal* · 🔑 `tickets.update`

1. Wählen Sie im Journal *Antwort an Melder* (im Portal: *Antworten*).
2. Schreiben Sie Ihre Nachricht.
3. Fügen Sie über *Textbausteine* bei Bedarf eine vorformulierte Antwort ein (Abschnitt 8).
4. Klicken Sie auf *Senden*.

✅ **Ergebnis:** Die Nachricht geht per E-Mail an den Melder und steht als Eintrag im Journal.

2.2 Interne Notiz

> 💻 **Client:** Ticket-Detail ▸ *Interne Notiz* · 🌐 **Portal:** Ticket-Detail ▸ Bearbeiter-Bereich ▸ *Interne Notiz* · 🔑 `tickets.update`

1. Wechseln Sie auf *Interne Notiz* (im Portal beschriftet *Nur für Bearbeiter sichtbar*).
2. Schreiben Sie Ihren Vermerk in das Feld *Notiz*.
3. Klicken Sie auf *Notiz speichern*.

✅ **Ergebnis:** Es erscheint *Notiz gespeichert.* Der Eintrag trägt im Journal die Marke *Intern*.

Eine interne Notiz geht **nie** per E-Mail hinaus und erscheint dem Melder auch im Portal nicht – die Schnittstelle liefert interne Einträge ausschliesslich an Bearbeiter aus.

> **Achtung:** „Intern" heißt intern gegenüber dem **Melder**, nicht gegenüber dem Haus. Andere Bearbeiter lesen die Notiz, und das Audit-Log hält sie fest. Was niemand lesen soll, gehört nicht in ein Ticket.

2.3 Zuweisen, Status, Priorität und Zeit erfassen

> 💻 **Client:** Ticket-Detail ▸ Info-Leiste · 🌐 **Portal:** Ticket-Detail ▸ Bereich *Bearbeitung* · 🔑 `tickets.update`

💡 **Empfohlene Reihenfolge** — zuerst zuweisen, dann den Status setzen. Andersherum steht der Vorgang kurzzeitig „in Bearbeitung" ohne Bearbeiter, und genau danach sucht die Warteschlange *Nicht zugewiesen*.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Wählen Sie unter *Bearbeiter* die zuständige Person – oder klicken Sie auf *Mir zuweisen*.
2. Wählen Sie bei Bedarf unter *Support-Gruppe* die zuständige Gruppe.
3. Setzen Sie *Status* auf den neuen Stand.
4. Passen Sie *Priorität* an, falls sich die Lage geändert hat (Abschnitt 3).
5. Tragen Sie unter *Zeiterfassung* die aufgewendeten *Minuten* mit *Datum* ein und klicken Sie auf *Zeit erfassen*.

✅ **Ergebnis:** Für jede Änderung erscheint eine kurze Rückmeldung (*Vorgang zugewiesen.*, *Status geändert.*, *Priorität geändert.*, *Zeit erfasst.*), und jede steht im Journal.

Vorgang zurückstellen: Setzen Sie *Zurückgestellt bis* auf ein Datum. Der Vorgang verschwindet damit aus *Offene Vorgänge* und taucht in der Warteschlange *Wiedervorlage* auf.

2.4 Eskalieren

> 💻 **Client:** Ticket-Detail ▸ *Aktionen* ▾ ▸ *An Vorgesetzten eskalieren* · 🌐 **Portal:** Ticket-Detail ▸ Bereich *Eskalation* · 🔑 `tickets.update`

1. Öffnen Sie den Bereich *Eskalation*.
2. Wählen Sie die *Eskalationsstufe*.
3. Tragen Sie unter *Begründung* ein, warum Sie eskalieren. Ohne Begründung ist eine Eskalation für die nächste Stelle nicht einzuordnen.
4. Klicken Sie auf *Eskalieren*.

✅ **Ergebnis:** Es erscheint *Vorgang eskaliert.* Die hinterlegte Stelle wird benachrichtigt, und die Stufe steht sichtbar am Vorgang.

2.5 Termin mit dem Melder vereinbaren

> 💻 **Nur im Client:** Ticket-Detail ▸ *Aktionen* ▾ ▸ *Termin vorschlagen* · 🔑 `tickets.update` — Im Portal sieht und beantwortet der **Melder** den Vorschlag (ID-10, Abschnitt 9); vorschlagen lässt er sich dort nicht.

Der Melder erhält eine Kalender-Einladung und sagt selbst zu oder ab, ganz ohne Anmeldung.

1. Klicken Sie oben im Ticket-Detail auf *Aktionen* ▾ (neben *Mir zuweisen*) und wählen Sie *Termin vorschlagen*.
2. Geben Sie *Betreff* und den *Zeitraum* (Beginn/Ende) an.
3. Tragen Sie bei Bedarf einen *Ort* ein.
4. Bestätigen Sie.

✅ **Ergebnis:** Der Melder bekommt eine **E-Mail mit Kalender-Einladung (.ics)** samt zwei Links: *Zusagen* oder *Absagen*. Ein Klick genügt – kein Konto nötig. Die Antwort wird **im Ticket-Journal** vermerkt. Bei **Zusage** wird der Termin Ihnen zusätzlich als **Wiedervorlage** registriert, damit er nicht untergeht.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> Die Schaltfläche *Aktionen* ▾ bündelt außerdem *An Vorgesetzten eskalieren*, *Zu Problem hochstufen*, *Checklisten* und *Ticket aufteilen* (Abschnitt 6.1). Sie ist sichtbar, wenn Sie das Ticket bearbeiten dürfen.

3 Kategorie, Dringlichkeit, Auswirkung & Priorität

> 💻 **Client:** Ticket-Detail ▸ Info-Leiste · 🌐 **Portal:** Ticket-Detail ▸ *Bearbeitung* · 🔑 `tickets.update`

- **Kategorie** ordnet das Ticket fachlich ein und kann Auto-Routing und SLA beeinflussen.
- **Dringlichkeit** und **Auswirkung** wählen Sie je als *Niedrig / Mittel / Hoch*.
- Die **Priorität** wird daraus **automatisch abgeleitet** (z. B. Hoch × Hoch → Kritisch).

So stufen Sie einen Vorgang richtig ein:

🔒 **Pflichtreihenfolge** — die Priorität ist ein Rechenergebnis aus den beiden Eingaben. Sie direkt setzen zu wollen führt nicht zum Ziel; sie wird beim nächsten Speichern überschrieben.

1. Prüfen Sie, **wie dringend** es ist – wie schnell muss es weg? Das ist die *Dringlichkeit*.
2. Prüfen Sie, **wen es trifft** – eine Person, eine Abteilung, das ganze Haus? Das ist die *Auswirkung*.
3. Setzen Sie beide Werte.

✅ **Ergebnis:** Die *Priorität* stellt sich selbst ein, und mit ihr die SLA-Fristen und die Eskalationskette.

> **Achtung:** „Kritisch" ist keine Eingabe, sondern ein Ergebnis. Setzen Sie Dringlichkeit und Auswirkung korrekt – daraus ergeben sich SLA-Fristen und Eskalationen.

4 SLA-Fristen & Eskalation

Jedes Ticket hat – je nach Priorität und SLA-Profil – eine **Reaktions-** und eine **Lösungsfrist**. Die verbleibende Zeit sehen Sie farblich hervorgehoben (im Portal als *Erste Reaktion* und *Reaktion fällig*). Läuft eine Frist ab, wird das Ticket **eskaliert** (z. B. Benachrichtigung an eine Führungskraft). Geschäftszeiten/Kalender bestimmen, welche Stunden gezählt werden.

> **Tipp:** Die Warteschlange *SLA verletzt* im Portal und die Spalte *SLA-Fälligkeit* im Client beantworten dieselbe Frage. Sehen Sie zu Dienstbeginn dort zuerst nach – eine gerissene Frist lässt sich am Vormittag noch erklären, am Abend nicht mehr.

5 ITIL-Prozesse im Alltag

Ordivis Platform kennt vier Ticket-Typen:

- **Incident** – Störung; Ziel ist die schnelle Wiederherstellung.
- **Service Request** – Standardauftrag; ggf. mit Genehmigung / Finanzstufe.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Problem** – Ursachenanalyse hinter wiederkehrenden Incidents; mündet in einen **Known Error**.
- **Change** – geplante Änderung; Genehmigung über das **CAB** (Change Advisory Board).

Details zu jedem Prozess stehen im Modul-Dokument ID-20.

6 Major Incident, Verknüpfen, Zusammenführen & Aufteilen

> 💻 **Nur im Client:** *Service Desk* ▸ *Tickets*, Werkzeugleiste und *Aktionen* ▾ · 🔑 `tickets.update` — Das Portal zeigt bestehende Verknüpfungen an, stellt sie aber nicht her.

- **Major Incident / Masterticket:** Bei einer großen Störung bündeln Sie viele Meldungen unter einem Master-Ticket.
- **Verknüpfen:** Stellen Sie Beziehungen zu anderen Tickets/CIs her.
- **Zusammenführen (Merge):** Doppelte Meldungen führen Sie zu einem Ticket zusammen.
- **Aufteilen (Split):** Das Gegenstück zum Zusammenführen – ein Thema aus einem Ticket wird ein eigener Vorgang (Abschnitt 6.1).

So bündeln Sie eine Großstörung:

🔒 **Pflichtreihenfolge** — erst das Masterticket bestimmen, dann die Meldungen anhängen. Ein nachträglicher Wechsel des Masters löst die bereits gesetzten Verknüpfungen nicht von selbst.

1. Öffnen Sie das Ticket, das die Störung am besten beschreibt.
2. Klicken Sie in der Werkzeugleiste auf *Major Incident / Masterticket*.
3. Markieren Sie in der Liste die doppelten Meldungen.
4. Klicken Sie auf *Zusammenführen* und bestätigen Sie.

✅ **Ergebnis:** Alle Melder hängen am selben Vorgang und erhalten dieselbe Rückmeldung – einmal geschrieben statt zwanzigmal.

6.1 Ticket aufteilen

> 💻 **Nur im Client:** Ticket-Detail ▸ *Aktionen* ▾ ▸ *Ticket aufteilen* · 🔑 `tickets.update`

Melder schreiben oft **mehrere Themen in eine E-Mail**. Fachlich sind das getrennte Vorgänge: andere Zuständigkeit, andere Priorität, eigene Frist. Solange sie in einem Ticket stecken, gilt der Vorgang erst als erledigt, wenn das **langsamste** Thema durch ist – und die Frist des schnellen wird an der Laufzeit des langsamen gemessen.

1. Öffnen Sie das Ticket und wählen Sie unter *Aktionen* ▾ den Punkt *Ticket aufteilen*.
2. Geben Sie den *Betreff* des abzuspaltenden Themas an – er ist das Einzige, was zwingend nötig ist, denn er *ist* das neue Thema. Ohne ihn bleibt die Schaltfläche deaktiviert.
3. Ergänzen Sie bei Bedarf eine *Beschreibung*. Sie ist freiwillig.
4. Wählen Sie bei Bedarf eine *Kategorie*. Lassen Sie sie leer, übernimmt der neue Vorgang die des Ursprungs – das ist der häufige Fall.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5. Bestätigen Sie. Danach können Sie direkt in den neuen Vorgang springen.

✅ **Ergebnis:** Es gibt zwei Vorgänge mit eigenen Fristen, wechselseitig verknüpft.

Was dabei geschieht:

Der Ursprung bleibt offen	Ob dort noch etwas zu tun ist, entscheiden Sie – nicht das System
Der Verlauf bleibt beim Ursprung	Dort stand die Mail mit *beiden* Themen. Der neue Vorgang ist **verknüpft** und zeigt darauf
Beide Vorgänge werden im Journal vermerkt	Wer den Ursprung ansieht, erkennt, dass ein Teil daraus jetzt woanders bearbeitet wird
Der neue Vorgang erhält eine eigene Frist	Genau das ist der Zweck der Aufteilung
Der Melder wird übernommen	Es ist dieselbe Person mit demselben Anliegen

Hinweis: Anhänge werden **nicht** mitkopiert. Sie bleiben beim Ursprung – über die Verknüpfung sind sie von beiden Seiten erreichbar, und es gibt sie nur einmal.

7 Wissen pflegen

💻 **Client:** Ticket-Detail ▸ *Artikel aus Lösung erstellen* · 🌐 **Portal:** *Wissen & Hilfe* ▸ *Wissensdatenbank-Redaktion* · 🔑 `knowledge.author`

Am Ticket erhalten Sie **Artikelvorschläge** aus der Wissensdatenbank. Aus einer guten Lösung machen Sie einen Artikel für die Zukunft:

💡 **Empfohlene Reihenfolge** — schreiben Sie den Artikel, solange der Fall frisch ist, und vor dem Schließen des Tickets. Danach fehlen erfahrungsgemäß die Einzelheiten, die ihn brauchbar machen.

1. Öffnen Sie das gelöste Ticket.
2. Klicken Sie auf *Artikel aus Lösung erstellen*.
3. Kürzen Sie den übernommenen Text auf das, was ein Unbeteiligter braucht – Symptom, Ursache, Lösungsweg.
4. Vergeben Sie Kategorie und Schlagworte, damit der Artikel gefunden wird.
5. Speichern Sie und geben Sie ihn nach dem Vier-Augen-Prinzip frei (ID-25).

✅ **Ergebnis:** Der nächste Melder mit demselben Problem bekommt den Artikel bereits beim Schreiben seines Tickets vorgeschlagen – und eröffnet im besten Fall gar keines.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8 Textbausteine (Canned Responses)

💻 **Client:** *Service Desk* ▸ *Textbausteine* · 🌐 **Portal:** *Service Desk* ▸ *Textbausteine und Vorlagen* · 🔑 `tickets.read` zum Lesen, `ticketing.config.manage` zum Ändern

Häufige Antworten hinterlegen Sie als **Textbausteine** und fügen sie mit einem Klick ein. Platzhalter (z. B. Melder-Name, Ticketnummer) werden automatisch ersetzt.

1. Öffnen Sie im Ticket das Antwortfeld.
2. Klicken Sie auf *Textbausteine*.
3. Wählen Sie den passenden Baustein aus der Liste.
4. Passen Sie den eingefügten Text an den Einzelfall an, bevor Sie senden.

✅ **Ergebnis:** Der Text steht im Antwortfeld, Platzhalter sind bereits ersetzt.

Hinweis: Steht *Für diesen Baustein gibt es hier kein Antwortfeld.*, haben Sie den Baustein außerhalb einer Antwort geöffnet. Öffnen Sie zuerst *Antworten*. Meldet das Portal *Keine Textbausteine hinterlegt.*, sind für Ihren Mandanten noch keine angelegt – die Verwaltung beschreibt ID-20.

9 Einen Fehler an den Support melden

💻 **Client:** Käfer-Symbol ganz links in der Statusleiste · 🌐 **Portal:** Käfer-Symbol in der Kopfleiste · 🔑 kein Recht nötig

Läuft die Anwendung auf einen Fehler, melden Sie ihn direkt aus dem Programm. Das Symbol ist auf jeder Seite an derselben Stelle und übermittelt die Seite, auf der Sie gerade stehen, als Zusammenhang mit.

🔒 **Pflichtreihenfolge** — der Datenschutzhinweis steht **vor** der Eingabe, und ohne gesetztes Häkchen wird nichts gelesen und nichts gesendet.

1. Klicken Sie auf das **Käfer-Symbol**.
2. Lesen Sie den **Datenschutzhinweis** im oberen Teil des Fensters.
3. Beschreiben Sie unter *Was haben Sie getan, und was ist passiert?*, was Sie erwartet haben und was stattdessen geschah. Ohne Beschreibung gibt es keinen Bericht.
4. Wählen Sie unter *Was soll mitgeschickt werden?* die Belege ab, die Sie nicht übermitteln möchten.
5. Klicken Sie auf *Einverstanden und senden*.

✅ **Ergebnis:** Es erscheint *Ihr Bericht wurde an den Support übermittelt.*

Abbildung 3 zeigt diesen Dialog im Client.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

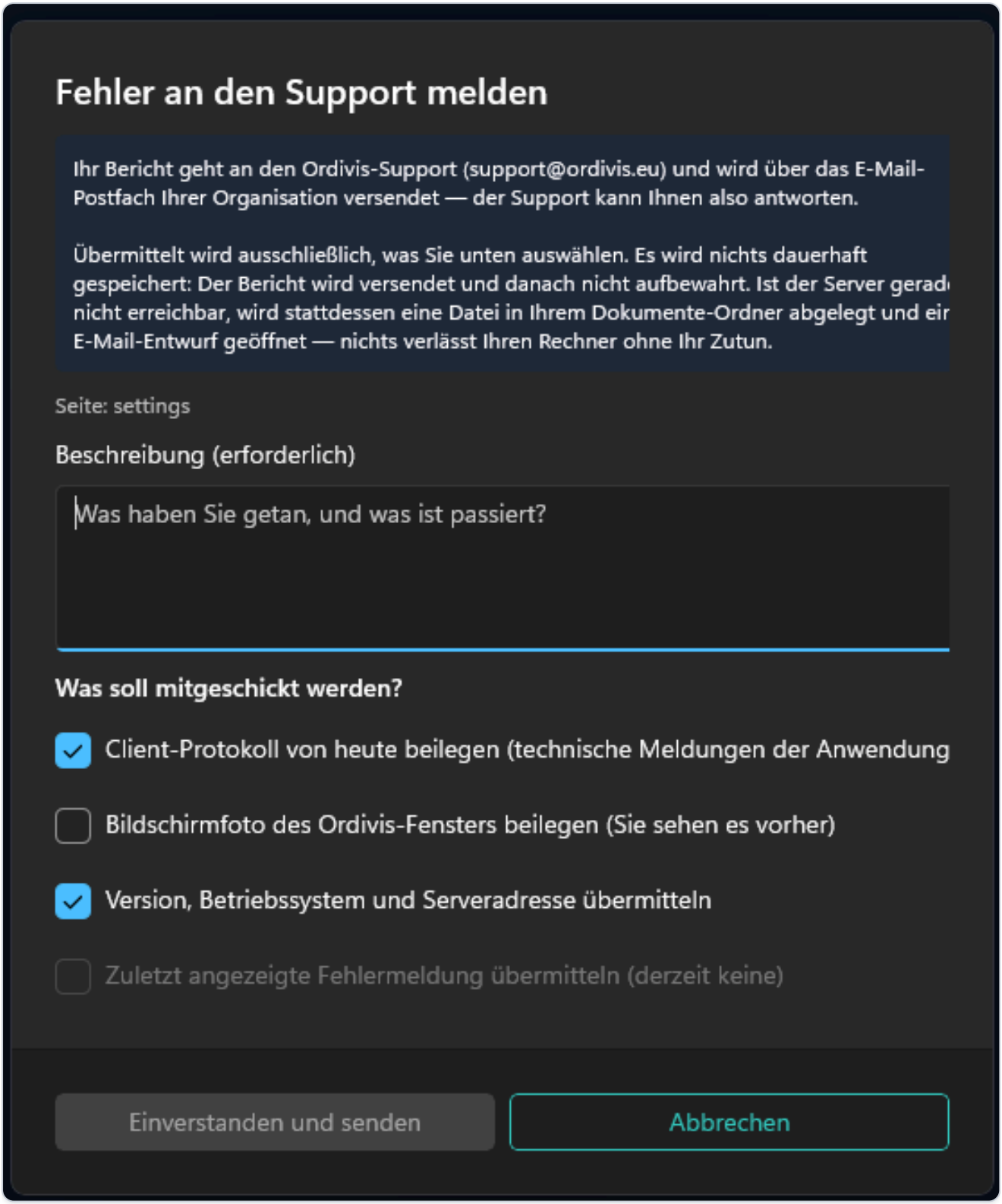

Abb. 3 – Fehler an den Support melden: Der Datenschutzhinweis steht *vor* der Eingabe, und jeder Beleg ist einzeln abwählbar. Das Bildschirmfoto ist voreingestellt abgeschaltet; nicht Verfügbares (hier „Zuletzt angezeigte Fehlermeldung“) ist ausgegraut statt stillschweigend weggelassen.

Was der Client mitschickt:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Beleg	Voreinstellung	Inhalt
Beschreibung	Pflicht	Ihre Schilderung – ohne sie kein Bericht
Client-Protokoll von heute	an	Technische Meldungen der Anwendung, nur vom laufenden Tag
Bildschirmfoto	**aus**	Nur das Ordivis-Fenster, nie der ganze Bildschirm
Systemdaten	an	Version, Betriebssystem, Serveradresse
Letzte Fehlermeldung	an, sofern vorhanden	Die zuletzt angezeigte Meldung

Was das Portal mitschickt: Der Browser kann kein Bildschirmfoto und kein Client-Protokoll liefern. Dort stehen deshalb nur zwei Häkchen zur Wahl – *Browser, Betriebssystem, Fenstergröße und Serveradresse übermitteln* sowie *Zuletzt angezeigte Fehlermeldung übermitteln*. Gibt es keine Fehlermeldung, steht das ausdrücklich daneben (*derzeit keine*), statt dass der Punkt fehlt.

Das Bildschirmfoto im Client ist bewusst **voreingestellt abgeschaltet** und wird Ihnen vor dem Versand angezeigt – darauf könnten Daten zu sehen sein, die niemanden außerhalb Ihres Hauses angehen. Sie können den Bericht an dieser Stelle noch ohne Bild abschicken.

Der Bericht geht über das E-Mail-Postfach **Ihrer Organisation** an den Support, damit dieser Ihnen antworten kann. Auf dem Server wird nichts davon aufbewahrt.

Unabhängig von den Häkchen enthält jeder Bericht, **wer** ihn gemeldet hat: Ihr Name und Ihre E-Mail-Adresse als Antwortadresse sowie Lizenznehmer und Mandant. Ohne diese Angaben wüsste der Support nicht, wem er antworten soll – die Antwort ginge an das Systempostfach Ihrer Organisation statt an Sie.

Wenn der Server nicht erreichbar ist – häufig genau dann, wenn Sie melden möchten – legt der Client den Bericht als ZIP-Datei unter *Dokumente\Ordivis-Fehlerberichte* ab und öffnet einen vorbereiteten E-Mail-Entwurf. Hängen Sie die Datei an und senden Sie sie ab. Ohne Ihr Zutun verlässt in diesem Fall nichts Ihren Rechner.

10 Tipps für effizientes Arbeiten

Tipp: Legen Sie sich im Client eine gespeicherte Ansicht „Meine offenen, nach Frist“ an (Abschnitt 1.2) – so sehen Sie auf einen Blick, was zuerst dran ist. Im Portal übernimmt das die Warteschlange *Mir zugewiesen*.

Tipp: Nutzen Sie interne Notizen für Zwischenstände. So bleibt der Verlauf für Kollegen nachvollziehbar, ohne den Melder mit Details zu belasten.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Tipp:** Erfassen Sie die Zeit sofort, nicht am Freitagnachmittag. Der Aufwand je Vorgang ist die Grundlage jeder späteren Aussage über Personalbedarf – nachträglich geschätzt ist er wertlos.

Verwandte Themen

- **ID-20 – Modul Service Desk / Ticketing** – Prozesse, SLA, Kategorien und Routing dahinter
- **ID-25 – Modul Wissensdatenbank** – Wissensartikel und bekannte Fehler pflegen
- **ID-34 – Modul Selbstorganisation** – eigene Aufgaben neben der Ticketarbeit organisieren
- **ID-27 – Modul Reporting & Dashboards** – Kennzahlen zur eigenen Arbeit
- **ID-10 – Benutzerhandbuch – Self-Service-Portal** – was der Melder auf der anderen Seite sieht

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

FÜR ALLE ANWENDER

Darstellung und Barrierefreiheit

Farbschema, Hell/Dunkel, Schriftgröße und Barrierefreiheit je Benutzer einstellen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-12 Darstellung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Darstellung und Barrierefreiheit

Ordivis Platform lässt sich je Benutzer an persönliche Seh- und Bediengewohnheiten anpassen. Alle Einstellungen wirken **sofort** (Live-Vorschau), werden dauerhaft gespeichert und gelten nur für Ihr Konto – im Windows-Client wie im Web-Portal. Niemand sonst sieht dadurch etwas anderes.

Client und Portal bieten nicht dasselbe an. Der Client hat die vollständige Auswahl auf einer Einstellungsseite; das Portal bringt eine kürzere Auswahl in einem Menü der Kopfleiste unter. Abschnitt 5 stellt beide gegenüber – dort steht auch, was es im Portal **nicht** gibt.

INHALT

1 Farbschema
1.1 Das Farbschema wechseln
2 Hell und Dunkel
3 Barrierefreiheit einstellen
4 Barrierefreiheit im Portal
5 Was gibt es wo? Client und Portal im Vergleich
6 Einheitliche Registerkarten
7 Technischer Hinweis (Pflege)
Verwandte Themen

1 Farbschema

Es stehen sieben aufeinander abgestimmte Farbschemata zur Verfügung, jeweils in einer hellen und einer dunklen Variante:

Schema	Charakter
Ordivis	Standard, Türkis und Navy des Corporate Designs
Blau	ruhiges Unternehmensblau
Türkis	frisch, kühl
Indigo	kräftig, modern
Neutral	zurückhaltendes Graublau
Warm	warmes Orange als Akzent
Hoher Kontrast	maximale Kontraste für Barrierefreiheit

Nach der Installation ist **Ordivis** eingestellt. Diese Vorgabe gilt so lange, bis der Benutzer selbst ein Schema wählt; ab dann bleibt seine Wahl bestehen und wird von Aktualisierungen nicht mehr angetastet.

Das Farbschema ändert nur die Akzentfarbe (Bedienelemente, Navigation). Die neutrale Flächen- und Textgestaltung bleibt konstant und ist durchgängig auf das WCAG-2.1-Niveau **AA** geprüft (Text-Kontrast

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

mindestens 4,5:1).

1.1 Das Farbschema wechseln

> 💻 **Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Erscheinungsbild* · 🌐 **Portal:** *Kopfleiste* ▸ *Paletten-Symbol* ▸ *Farbschema* · 🔑 kein Recht nötig

Im Client:

1. Klicken Sie unten im Menü auf *Einstellungen*.
2. Wechseln Sie in den Bereich *Darstellung*.
3. Öffnen Sie die Auswahlliste *Farbschema*. Jeder Eintrag zeigt links ein Farbplättchen.
4. Wählen Sie ein Schema aus.

Im Portal:

1. Klicken Sie oben rechts auf das **Paletten-Symbol**.
2. Wählen Sie unter *Farbschema* den gewünschten Eintrag. Das aktive Schema trägt ein Häkchen.

✅ **Ergebnis:** Die Oberfläche färbt sich sofort um – ein Speichern-Klick ist nicht nötig, und ein Neustart auch nicht.

2 Hell und Dunkel

Unabhängig vom Schema lässt sich zwischen **Hell**, **Dunkel** und **Systemstandard** (folgt der Windows- bzw. Browser-Einstellung) wählen.

> 💻 **Client:** Sonne-/Mond-Symbol rechts in der Kopfleiste · 🌐 **Portal:** Sonne-/Mond-Symbol rechts in der Kopfleiste, neben dem Paletten-Symbol · 🔑 kein Recht nötig

1. Klicken Sie auf das **Sonne-/Mond-Symbol** in der Kopfleiste.

✅ **Ergebnis:** Die Ansicht wechselt zwischen hell und dunkel. Das Symbol zeigt immer das, was Sie mit dem nächsten Klick bekommen.

> **Hinweis:** Beim ersten Start folgt Ordivis Platform Ihrer System- bzw. Browsereinstellung. Sobald Sie einmal umgeschaltet haben, gilt Ihre Wahl – auch dann, wenn Windows später wechselt.

3 Barrierefreiheit einstellen

> 💻 **Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Erscheinungsbild* · 🔑 kein Recht nötig

Der Client führt fünf Einstellungen in einer eigenen Karte *Barrierefreiheit* zusammen:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Einstellung	Wirkung
Hoher Kontrast	Schaltet auf das Hochkontrast-Schema. Ein systemseitig aktivierter Windows-Hochkontrastmodus wird ohne Zutun übernommen.
Bewegung reduzieren	Unterdrückt Animationen und weiche Übergänge. Die Systemeinstellung `prefers-reduced-motion` wird ebenfalls beachtet.
Links unterstreichen	Blendet Unterstreichungen für Verweise dauerhaft ein – hilfreich, wenn Farbe allein den Link nicht erkennbar macht.
Fokus-Rahmen immer sichtbar	Hebt das per Tastatur angesteuerte Element deutlich hervor. Wer ohne Maus arbeitet, sollte das einschalten.
Schriftgröße	Schieberegler von **80 % bis 150 % in Schritten von 10 %**.

So stellen Sie sie ein:

1. Klicken Sie unten im Menü auf *Einstellungen*.
2. Wechseln Sie in den Bereich *Darstellung* und suchen Sie die Karte *Barrierefreiheit*.
3. Schalten Sie die gewünschten Schalter auf *Ein*.
4. Ziehen Sie den Regler *Schriftgröße* auf den gewünschten Wert.

✅ **Ergebnis:** Jede Änderung wirkt sofort in der laufenden Ansicht.

> **Wichtig:** Die vergrößerte Schrift greift **sofort in neu geöffneten Ansichten**. Bereits geöffnete Seiten übernehmen sie vollständig erst nach einem Neustart des Clients. Wundern Sie sich also nicht, wenn eine Liste zunächst unverändert aussieht.

4 Barrierefreiheit im Portal

> 🌐 **Nur im Portal:** *Kopfleiste* ▸ *Paletten-Symbol* ▸ *Barrierefreiheit* · 🔑 kein Recht nötig

1. Klicken Sie oben rechts auf das **Paletten-Symbol**.
2. Klicken Sie unter *Barrierefreiheit* auf *Hoher Kontrast* oder *Bewegung reduzieren*. Ein angehaktes Kästchen zeigt, dass die Einstellung aktiv ist; ein erneuter Klick schaltet sie ab.
3. Für die Schriftgröße klicken Sie auf *Schrift vergrößern* oder *Schrift verkleinern* – jeder Klick ändert um 10 Prozentpunkte. *Schriftgröße zurücksetzen* stellt 100 % wieder her.

✅ **Ergebnis:** Die Seite passt sich sofort an. Die Wahl wird zu Ihrem Konto gespeichert und gilt auch auf einem anderen Rechner.

> **Ebenfalls in diesem Menü: *Wetter im Menü zeigen*.** Der Schalter blendet den Menüpunkt *Wetter* aus, wenn er Sie nicht betrifft. Er ist eine Aufräum-, keine Sperrfunktion: Die Wetterseite bleibt über ihre Adresse erreichbar.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5 Was gibt es wo? Client und Portal im Vergleich

Einstellung	💻 Client	🌐 Portal
Farbschema (7 Schemata)	**ja** – Auswahlliste in den Einstellungen	**ja** – Paletten-Menü
Hell / Dunkel	**ja** – Kopfleiste	**ja** – Kopfleiste
Hoher Kontrast	**ja**	**ja**
Bewegung reduzieren	**ja**	**ja**
Schriftgröße	**ja** – Regler, 80–150 % in 10er-Schritten	**ja** – drei Menüpunkte, je 10 Prozentpunkte
Links unterstreichen	**ja**	–
Fokus-Rahmen immer sichtbar	**ja**	–
Wetter im Menü zeigen	–	**ja**

Wichtig: *Links unterstreichen* und *Fokus-Rahmen immer sichtbar* gibt es im Portal **nicht**. Wer auf beides angewiesen ist, arbeitet im Client. Für den Fokus-Rahmen gilt im Browser die Einstellung des Browsers selbst.

6 Einheitliche Registerkarten

Registerkarten (Tabs) sind in der gesamten Anwendung einheitlich gestaltet – als abgesetzte, gruppierte Fläche mit hervorgehobener aktiver Karte. Maßstab ist die Darstellung der Registerkarten in der **Prozessdokumentation** (Steckbrief / BPMN / Dokumente); alle weiteren Tab-Ansichten folgen diesem Standard.

7 Technischer Hinweis (Pflege)

Die Farbschemata stammen aus einer zentralen, maschinenlesbaren Quelle (`design/theme-schemes.json`). Der Generator `design/gen_theme.py` erzeugt daraus die Farbtabellen für Client und Portal und prüft dabei alle Rollenpaare gegen WCAG AA (`python design/gen_theme.py --check` – Fehlercode bei Kontrastverstoß, als Regressions-Gate geeignet). Generierte Dateien nicht von Hand ändern, sondern die Quelle anpassen und neu generieren.

Verwandte Themen

- **ID-10 – Benutzerhandbuch – Self-Service-Portal** – das Portal aus Sicht der Endbenutzer
- **ID-11 – Agentenhandbuch – Service Desk** – der Arbeitsplatz der Agenten
- **ID-40 – Administrationshandbuch** – mandantenweites Branding statt persönlicher Einstellung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-01 – Einführung & Grundlagen** – Abschnitt 3.3 stellt beide Oberflächen gegenüber

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · SERVICE DESK

Modul Service Desk / Ticketing

Ticketing nach ITIL – Prozesse, SLA, Kategorien, Routing, Konfiguration

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-20 Service Desk

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Service Desk / Ticketing

Wenn im Unternehmen etwas mit der IT nicht stimmt – der Drucker streikt, ein Programm hängt, jemand braucht einen neuen Zugang –, muss die Meldung irgendwo landen, wo sie nicht verloren geht. Zuruf über den Flur, eine E-Mail hier, ein Anruf dort: So gehen Anliegen unter, niemand weiß, wer sich kümmert, und der Melder erfährt nie, was aus seiner Sache wurde.

Ein Service Desk ist die zentrale Anlaufstelle, die das ordnet. Jede Meldung wird zu einem *Ticket*, also einem nachvollziehbaren Vorgang mit Nummer, Verlauf und Zuständigem. Ordivis Platform unterscheidet dabei vier Arten von Vorgängen, wie sie das IT-Rahmenwerk ITIL vorsieht: die Störung (Incident), den Standardauftrag (Service Request), die Ursachensuche (Problem) und die geplante Änderung (Change). Dazu kommen vereinbarte Bearbeitungszeiten (SLA), Eskalationen bei Verzug und Genehmigungen. Dieses Dokument beschreibt beides: die tägliche Nutzung und die administrative Einrichtung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

1 Zweck & Nutzen
2 Voraussetzungen
3 Ticket-Typen (ITIL)
3.1 Vorgangsnummern
4 Ticket-Lebenszyklus
4.1 Melder nachtragen oder korrigieren
4.2 Antwort des Melders holt das Ticket zurück in die Bearbeitung
4.3 Betroffene Konfigurationselemente
4.4 Interessierter Kreis
4.5 Entscheider
5 Priorität aus Dringlichkeit × Auswirkung
6 SLA, OLA & Eskalation
6.1 Servicezeiten pflegen
6.2 Eskalationsstufen: wann die nächste Stelle erfährt
6a Termine am Ticket: Fälligkeit, Zurückstellung, Wiedervorlage
6a.1 Fälligkeit von Hand setzen
6a.2 Zurückstellen
6a.3 Wiedervorlage
6a.4 Erinnerungen
6a.5 Termine in der Ticketliste und in „Meine Termine"
7 Support-Gruppen & Sichtbarkeit
8 Auto-Routing
8.1 Kategorie aus dem Betreff setzen
9 Genehmigungen & Finanzstufen (Service Request)
10 Problem-Management & KEDB
11 Change-Management & CAB
11.1 Einen RFC anlegen und einreichen
12 Kundenzufriedenheit (CSAT)
12a Arbeiten in der Ticketliste
12b Bilder und Dateien ins Journal einfügen
13 Konfiguration (Administration)
14 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Einheitlicher Eingangskanal für Anwenderanliegen (Portal, E-Mail, Agent).
- Nachvollziehbare Bearbeitung mit vollständigem Journal.
- Einhaltung von Service-Zeiten über SLA/OLA und Eskalation.
- Wiederverwendbares Wissen (KEDB, Wissensdatenbank).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2 Voraussetzungen

- Mindestens ein **SLA-Profil**, ein **Geschäftszeiten-Kalender** und ein **Kategorien-Baum**.
- Für E-Mail-Eingang/-Versand: konfigurierte E-Mail-Integration (ID-31).
- Berechtigungen: `tickets.read`, `tickets.create`, `tickets.update`, ggf. `tickets.read.all` (siehe ID-41).

3 Ticket-Typen (ITIL)

Typ	Zweck	Besonderheit
Incident	Störung beheben	schnelle Wiederherstellung, ggf. Major Incident
Service Request	Standardleistung	Genehmigungen, Finanzstufen
Problem	Ursachenanalyse	Root-Cause, **Known Error** (KEDB)
Change	geplante Änderung	Risikobewertung, **CAB**-Freigabe

3.1 Vorgangsnummern

Jeder Vorgang bekommt beim Anlegen eine Nummer aus Kürzel, Jahr und laufender Zahl – etwa `INC-20260042`. Die Zählung läuft **je Mandant, Vorgangsart und Jahr** und beginnt mit jedem Jahreswechsel neu. Probleme (`PRB-…`) und Änderungen (`RFC-…`) haben eine eigene, durchlaufende Zählung ohne Jahresanteil.

> **Die Nummern können Lücken haben.** Wird ein Anlegen abgebrochen oder scheitert es, bleibt die bereits vergebene Nummer verbraucht. Das ist beabsichtigt: Nur so kann die Nummer bei mehreren gleichzeitigen Anlagen nicht zweimal vergeben werden. Eine fehlende Nummer bedeutet also **nicht**, dass ein Vorgang gelöscht wurde – Vorgänge werden in Ordivis Platform ohnehin geschlossen und nicht entfernt.

4 Ticket-Lebenszyklus

Ein Ticket durchläuft typischerweise: **Neu** → **Zugewiesen** → **In Bearbeitung** → **Wartet** → **Gelöst** → **Geschlossen**. Statuswechsel werden im Journal protokolliert; bestimmte Übergänge lösen Benachrichtigungen und SLA-Ereignisse aus. Abbildung 1 zeigt den Ablauf im Zusammenhang.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

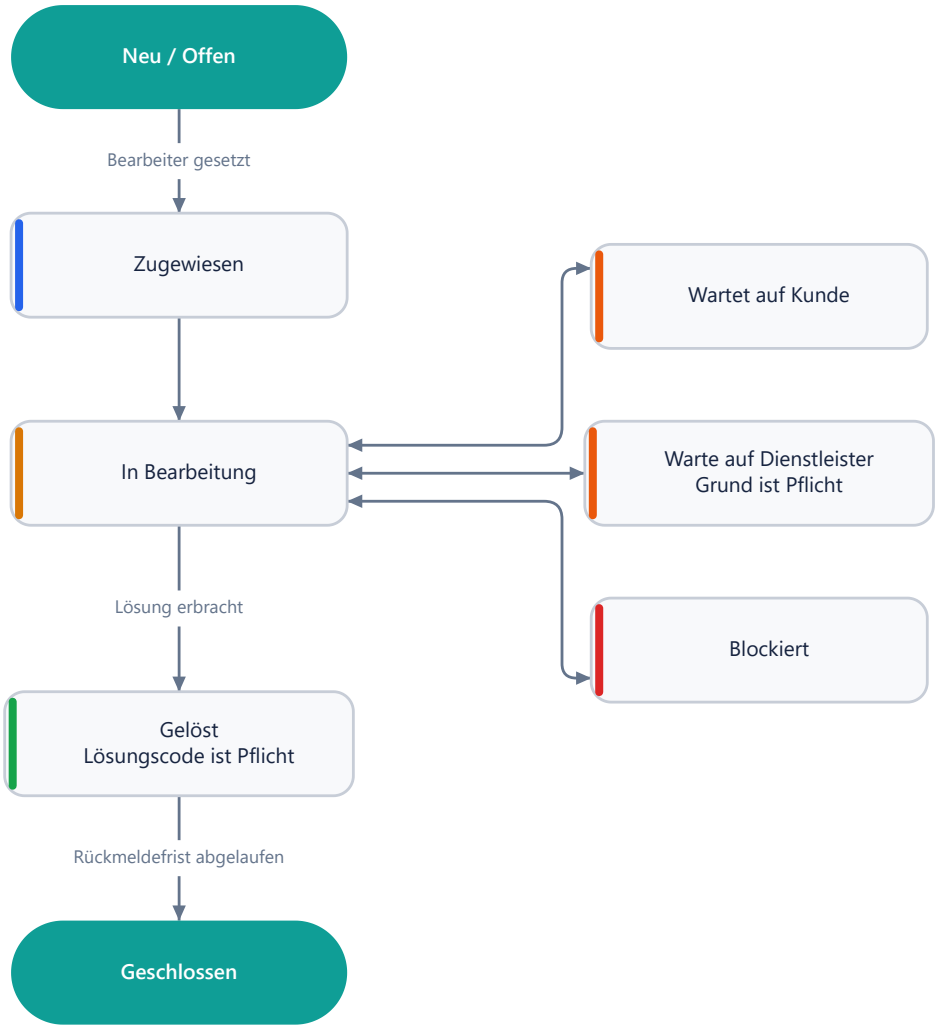

Abb. 1 – Ticket-Lebenszyklus. Die beidseitigen Pfeile bedeuten: Aus jedem Warte-Zustand kehrt der Vorgang in die Bearbeitung zurück.

Es gibt drei unterschiedliche Warte-Zustände – sie sind bewusst getrennt, weil sie verschiedene Zuständigkeiten bedeuten. In allen dreien **pausiert die SLA-Uhr**; beim Verlassen wird die Frist um die Wartezeit verlängert.

Status	Farbe	Bedeutung
Neu / Offen	Blau	Noch nicht in Bearbeitung genommen
In Bearbeitung	Gelb	Wird aktiv bearbeitet
Wartet auf Kunde	Orange	Rückmeldung des Melders steht aus
Warte auf Dienstleister	Orange	Zuarbeit eines Dritten steht aus (Hersteller, Bestellung); **Grund ist Pflicht**
Blockiert	Rot	Intern blockiert – fehlende Zuarbeit, Abhängigkeit oder Entscheidung
Gelöst	Grün	Lösung erbracht, Rückmeldefrist läuft; **Lösungscode ist Pflicht**
Geschlossen	Grau	Abgeschlossen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Dieselben Farben verwenden die Statusspalte der Ticketliste, die Ticketansicht und die Legende der Liste – sie stammen aus einer gemeinsamen Quelle und können nicht auseinanderlaufen.

4.1 Melder nachtragen oder korrigieren

Ist der **Melder** eines Tickets zunächst unbekannt – etwa bei einem Anruf oder einer Mail aus einem Sammelpostfach – lässt er sich **nachträglich setzen**. Sobald ein Melder eingetragen ist, bleibt er über die Verknüpfung mit dem Kontakt gepflegt; Name und Erreichbarkeit stammen weiterhin aus dem Kontaktverzeichnis und werden nicht ins Ticket kopiert.

Bei Tickets **aus E-Mails** wertet Ordivis Platform zusätzlich die **Antwort-Adresse (Reply-To)** aus. So wird der tatsächliche Kunde als Melder erkannt und nicht das Postfach, das die Mail weitergeleitet hat – Rückfragen erreichen damit die richtige Person.

4.2 Antwort des Melders holt das Ticket zurück in die Bearbeitung

Antwortet der Melder per E-Mail auf ein Ticket, wird seine Nachricht nicht nur ins Journal aufgenommen – das Ticket **wechselt auch den Status**:

Status vor der Antwort	Status danach
Wartet auf Kunde · Warte auf Dienstleister	In Bearbeitung
Gelöst · Geschlossen	In Bearbeitung (**Wiedereröffnung**)
Neu · In Bearbeitung · Blockiert	unverändert

Abbildung 2 stellt dieselbe Regel als Entscheidung dar.

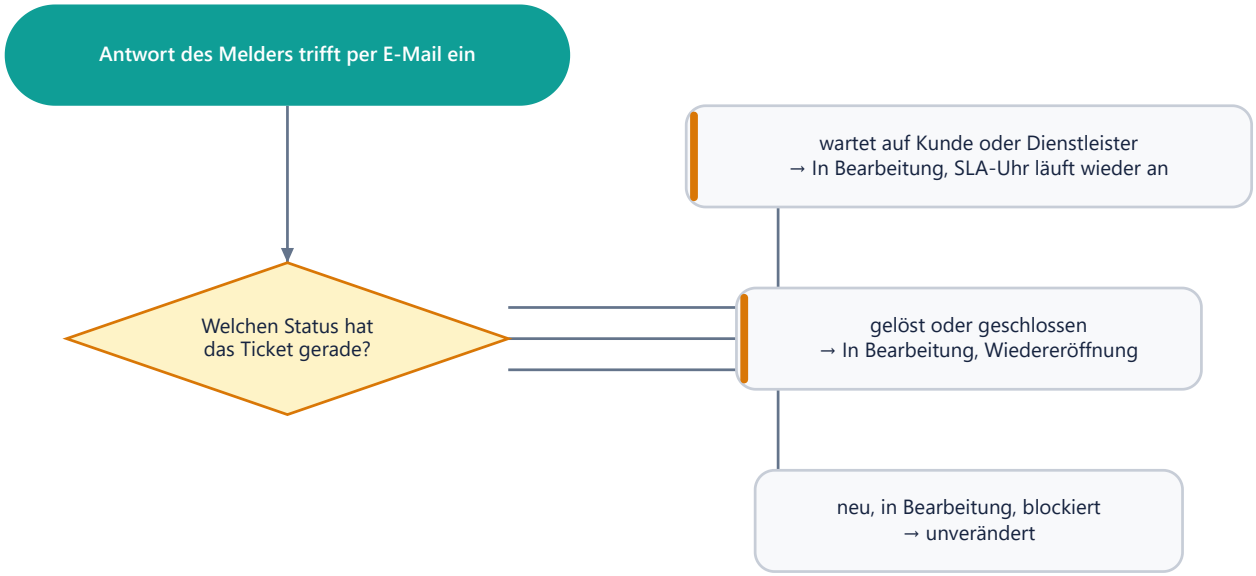

Abb. 2 – Was die Antwort des Melders am Status ändert. Entscheidend ist der Status, den das Ticket beim Eintreffen der Nachricht hat.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

„Blockiert" bleibt bewusst stehen: Ein interner Blocker wird durch eine Nachricht des Melders nicht aufgehoben. Solche Tickets holt der Bearbeiter selbst zurück in die Bearbeitung.

Beim Verlassen eines Warte-Status läuft die **SLA-Uhr wieder an**; eine Wiedereröffnung aus „Gelöst" oder „Geschlossen" zählt als solche und setzt Lösungs- bzw. Abschlusszeitpunkt zurück. Der Wechsel wird als Systemeintrag im Journal protokolliert.

4.3 Betroffene Konfigurationselemente

Ein Ticket führt ein **Haupt-CI** (das betroffene Konfigurationselement) sowie beliebig viele **weitere betroffene CIs**. Die Unterscheidung ist bewusst: SLA-Zuordnung, Reporting und die CMDB-Ausfallanalyse werten das Haupt-CI aus, während die weiteren CIs den tatsächlichen Umfang einer Störung dokumentieren. Beide pflegen Sie in der Seitenspalte des Tickets unter **Assets**; dasselbe CI lässt sich nicht doppelt verknüpfen.

4.4 Interessierter Kreis

Neben Melder und Bearbeiter kann ein Ticket weitere **Beteiligte** führen – Personen, die den Vorgang mitverfolgen sollen. Sie werden in der Seitenspalte unter **Interessierter Kreis** eingetragen.

Beteiligte erhalten dieselben E-Mails wie der Melder. Das gilt für interne Benutzer wie für externe Adressen gleichermaßen; interne Beteiligte bekommen zusätzlich eine Meldung in der Anwendung. Konkret:

Ereignis	Beteiligte werden benachrichtigt
Kundengerichtete Antwort	ja
Statuswechsel (Kapitel 4 – *Ticket-Lebenszyklus*)	ja
Übergabe an einen Bearbeiter	ja
Datei am Vorgang hinzugefügt	ja
Termin vorgeschlagen, zu- oder abgesagt	ja
Interne Notiz	**nein** – sie verlässt den Service Desk nicht

Beim Eintragen erhält ein Beteiligter **sofort eine E-Mail mit dem bisherigen Stand** des Vorgangs. Er steigt damit auf demselben Wissensstand ein wie der Melder und muss nicht nachfragen, worum es geht.

> **Statuswechsel:** Nach außen gemeldet werden nur Wechsel, die für einen Melder eine Bedeutung haben – *In Bearbeitung*, *Wartet auf Kunde*, *Gelöst*, *Geschlossen*. Interne Wartezustände (*Warte auf Dienstleister*, *Blockiert*) lösen **keine** E-Mail aus; der Bearbeiter sieht sie weiterhin vollständig. Ohne diese Grenze erzeugte allein das interne Hin und Her mehrere Mails je Vorgang.

Jede dieser E-Mails enthält den **vollständigen Vorgang**: Kopfdaten, Beschreibung, den öffentlichen Verlauf und die Liste der Anhänge. Der Empfänger muss sich nicht anmelden, um die Nachricht zu verstehen – was für externe Beteiligte ohnehin nicht möglich wäre. **Interne Notizen sind darin nie enthalten.**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Eintragen lassen sich Benutzer aus der Suche sowie freie E-Mail-Adressen. **Gruppen sind hier nicht wählbar**: Eine Benachrichtigung braucht einen konkreten Empfänger; tragen Sie die betreffenden Personen einzeln ein.

4.5 Entscheider

> 💻 **Nur im Client:** Ticket-Detail ▸ Seitenspalte ▸ *Entscheider* · 🔑 `tickets.update` — Das Portal zeigt den eingetragenen Entscheider an, versendet aber keine Entscheidungsanfrage.

Braucht ein Vorgang eine Entscheidung außerhalb des Service Desks, tragen Sie einen **Entscheider** ein. Er erhält eine E-Mail mit dem vollständigen Vorgang und kann direkt darauf antworten; wie beim interessierten Kreis bleiben interne Notizen außen vor.

🔒 **Pflichtreihenfolge** — die Anfrage geht an die hinterlegte Adresse. Ohne Auswahl aus der Personensuche (Schritt 2) gibt es keine, und Schritt 3 läuft ins Leere.

1. Öffnen Sie das Ticket und suchen Sie in der Seitenspalte das Feld *Entscheider*.
2. Wählen Sie die Person **aus der Personensuche** aus.
3. Klicken Sie auf *Entscheidungsanfrage senden*.

✅ **Ergebnis:** Der Entscheider bekommt den Vorgang per E-Mail und kann unmittelbar antworten; die Antwort landet im Journal.

> **Wählen Sie den Entscheider aus der Personensuche.** Nur dann ist seine Adresse hinterlegt und er kann benachrichtigt werden. Ein von Hand eingetippter Eintrag („Amtsleitung") bleibt zulässig und ist dann ein reiner Vermerk am Vorgang – die Oberfläche weist mit dem Hinweis *„Nicht benachrichtigbar"* darauf hin, bevor Sie auf Senden klicken.

5 Priorität aus Dringlichkeit × Auswirkung

Ordivis Platform leitet die Priorität aus einer Matrix ab:

Auswirkung ↓ / Dringlichkeit →	Niedrig	Mittel	Hoch
Hoch	Mittel	Hoch	Kritisch
Mittel	Niedrig	Mittel	Hoch
Niedrig	Niedrig	Niedrig	Mittel

> **Achtung:** Als Eingabewerte existieren nur **Niedrig/Mittel/Hoch**. „Kritisch“ ist ausschließlich eine abgeleitete Priorität – nicht direkt setzbar.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6 SLA, OLA & Eskalation

Abbildung 3 zeigt ein SLA-Profil mit seinen Fristen je Priorität.

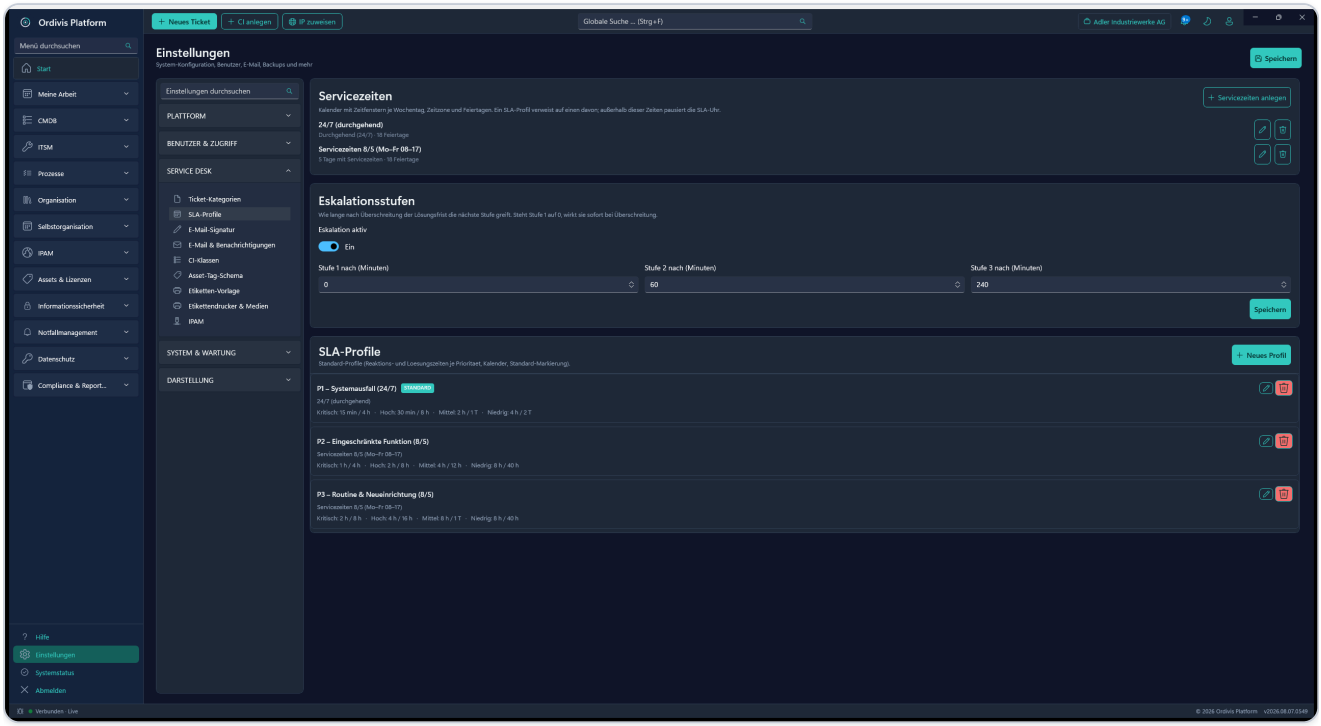

Abb. 3 – SLA-Profile & Servicezeiten: je Profil die Reaktions- und Lösungsfristen pro Priorität, wahlweise Business-Hours oder 24/7.

- **SLA-Profil:** definiert je Priorität die Reaktions- und Lösungsfrist.
- **Geschäftszeiten/Kalender:** legen fest, welche Stunden zählen (inkl. Feiertage).
- **OLA:** interne Zulieferzeiten zwischen Support-Gruppen.
- **Eskalation:** Bei Fristüberschreitung greifen Eskalationsstufen (Benachrichtigung, Umleitung).
- **Wartungsfenster:** setzen SLA für geplante Ausfälle aus.

6.1 Servicezeiten pflegen

> 🖥️ **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *SLA-Profile*, Karte *Servicezeiten* · 🔑 `system.config.manage` — Das Portal führt die Service-Desk-Konfiguration unter *Verwaltung* ▸ *Service-Desk-Konfiguration*, aber ohne die Servicezeiten-Kalender.

Ein Kalender führt:

- **Zeitfenster je Wochentag** – ein leeres Feld bedeutet: an diesem Tag kein Service.
- **Zeitzone** – maßgeblich für die Umrechnung der Fristen.
- **Durchgehend (24/7)** – die Uhr läuft ohne Unterbrechung; Zeitfenster spielen dann keine Rolle.
- **Feiertage** – Tage, an denen die Uhr steht.

Jedes SLA-Profil verweist auf **einen** Kalender. Außerhalb der hinterlegten Zeiten pausiert die SLA-Uhr und läuft danach weiter; eine Frist verfällt also nicht über Nacht oder am Wochenende.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ein Kalender braucht Geschäftszeit. Ein Kalender, der *nicht* auf **Durchgehend (24/7)** steht, muss mindestens ein Zeitfenster führen – sonst weist er zu keiner Sekunde Servicezeit aus, und es gäbe keinen Zeitpunkt, an dem eine Frist ablaufen könnte. Das Speichern wird in diesem Fall mit einem Hinweis abgelehnt. Wenn Sie einen Kalender anlegen und die Zeiten erst später pflegen wollen, setzen Sie ihn so lange auf **Durchgehend (24/7)** oder tragen ein vorläufiges Fenster ein.

So tragen Sie die Feiertage eines Bundeslandes ein:

🔒 **Pflichtreihenfolge** — ein erneuter Lauf **ersetzt** den gewählten Jahrgang, er ergänzt ihn nicht. Von Hand nachgetragene Gemeinde- oder Betriebsruhetage desselben Jahres sind danach weg. Berechnen Sie deshalb zuerst, ergänzen Sie danach.

1. Öffnen Sie den Bearbeiten-Dialog des Kalenders und wechseln Sie zu *Feiertage*.
2. Wählen Sie das **Bundesland**.
3. Wählen Sie das **Jahr**.
4. Bestätigen Sie mit *Übernehmen*.
5. Wiederholen Sie das für jedes Jahr, das Sie brauchen – üblicherweise das laufende und das kommende.

✅ **Ergebnis:** Die Anzahl in der Überschrift zeigt die eingetragenen Tage. Ordivis Platform **berechnet** die gesetzlichen Feiertage; ein Internetzugang ist dafür nicht nötig, was für Arbeitsplätze ohne Internet wichtig ist.

Nicht enthalten sind Feiertage, die nur in einzelnen **Gemeinden** gelten (etwa Mariä Himmelfahrt in überwiegend katholischen bayerischen Gemeinden oder Fronleichnam in Teilen von Sachsen und Thüringen). Sie hängen von der Gemeinde ab und lassen sich nicht landesweit ableiten – solche Tage sowie Betriebsruhetage tragen Sie von Hand nach.

Einzelnen Tag nachtragen. Unterhalb der Feiertagsliste steht dafür eine Zeile mit Datum, Bezeichnung und der Möglichkeit, den Tag als **halben Tag** zu kennzeichnen. Das ist der übliche Fall für **Heiligabend und Silvester**, an denen in vielen Verwaltungen und Betrieben ab mittags dienstfrei ist. Geben Sie dazu die Uhrzeit an, **ab der** die Servicezeit endet (etwa 13:00): Bis dahin läuft die SLA-Uhr normal, danach steht sie wie an einem ganzen Feiertag. Die Uhrzeit ist Pflicht – ohne sie wäre der Tag „halb frei ab unbestimmt", und die Fristberechnung müsste raten. Von Hand eingetragene Tage bleiben beim Bundesland-Import erhalten, solange sie nicht im selben Jahrgang liegen (der Import **ersetzt** den gewählten Jahrgang vollständig – tragen Sie eigene Tage also danach ein).

Ein Kalender, auf den ein SLA-Profil verweist, lässt sich **nicht löschen**; die Meldung nennt die betroffenen Profile. Das ist Absicht: Ein Profil ohne Kalender würde stillschweigend rund um die Uhr rechnen und damit alle Fristen verschärfen, ohne dass es jemandem auffiele.

6.2 Eskalationsstufen: wann die nächste Stelle erfährt

Wird die **Lösungsfrist** überschritten, hebt Ordivis Platform die Eskalationsstufe des Vorgangs an und benachrichtigt den für diese Stufe hinterlegten Empfänger der zuständigen Support-Gruppe (Stufe 1 und 2 sowie eine Stufe für die Leitung, siehe *Support-Gruppen*).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *SLA-Profile*, Karte *Eskalationsstufen* · 🔑 `system.config.manage`

Eingetragen wird je Stufe, **wie viele Minuten nach Überschreitung** sie greift:

Einstellung	Voreinstellung	Bedeutung
Stufe 1 nach	0 min	sofort mit der Überschreitung
Stufe 2 nach	60 min	eine Stunde später
Stufe 3 nach	240 min	vier Stunden später

Die Voreinstellung folgt der verbreiteten Praxis, Eskalationen in Stundenschritten weiterzureichen – erst im Team, dann in der Bereichs-, dann in der Hausleitung.

So passen Sie die Stufen an:

🔒 **Pflichtreihenfolge** — die Stufen müssen **aufsteigend** bleiben. Tragen Sie die Werte von Stufe 1 nach Stufe 3 ein; wer mittendrin einen kleineren Wert setzt, bekommt das Speichern abgelehnt.

1. Öffnen Sie *Einstellungen* ▸ *SLA-Profile* und suchen Sie die Karte *Eskalationsstufen*.
2. Tragen Sie bei *Stufe 1 nach* die Minuten ein, nach denen die erste Stelle erfährt.
3. Tragen Sie bei *Stufe 2 nach* einen **größeren** Wert ein.
4. Tragen Sie bei *Stufe 3 nach* einen nochmals größeren Wert ein.
5. Speichern Sie.

✅ **Ergebnis:** Die Kette greift ab der nächsten Fristüberschreitung. Ordnen Sie die Werte so an, wie es Ihre Dienstvereinbarung oder Ihr Vertrag vorsieht.

Achtung: Wären die Stufen nicht aufsteigend, erführe die Leitung von einer Überschreitung vor dem eigenen Team. Deshalb wird das Speichern in dem Fall abgelehnt – die Meldung nennt die betroffene Stufe.

Mit dem Schalter **Eskalation aktiv** lässt sich das Anheben ganz abschalten. Fristüberschreitungen werden dann weiterhin erkannt und protokolliert, es wird nur niemand zusätzlich benachrichtigt.

Je Überschreitung eine Stufe je Prüflauf. Die Prüfung läuft minütlich, hebt aber höchstens eine Stufe an – und erst, wenn deren Wartezeit abgelaufen ist. Ohne diese Zeitachse erreichte ein überfälliger Vorgang binnen drei Minuten die höchste Stufe und löste drei Meldungen aus.

Unberührt bleiben die **Wartungsfenster**: Ein Vorgang, der in ein laufendes Wartungsfenster fällt, wird nicht als verletzt gewertet und deshalb auch nicht eskaliert. Abbildung 4 zeigt, in welcher Reihenfolge diese drei Bedingungen geprüft werden.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

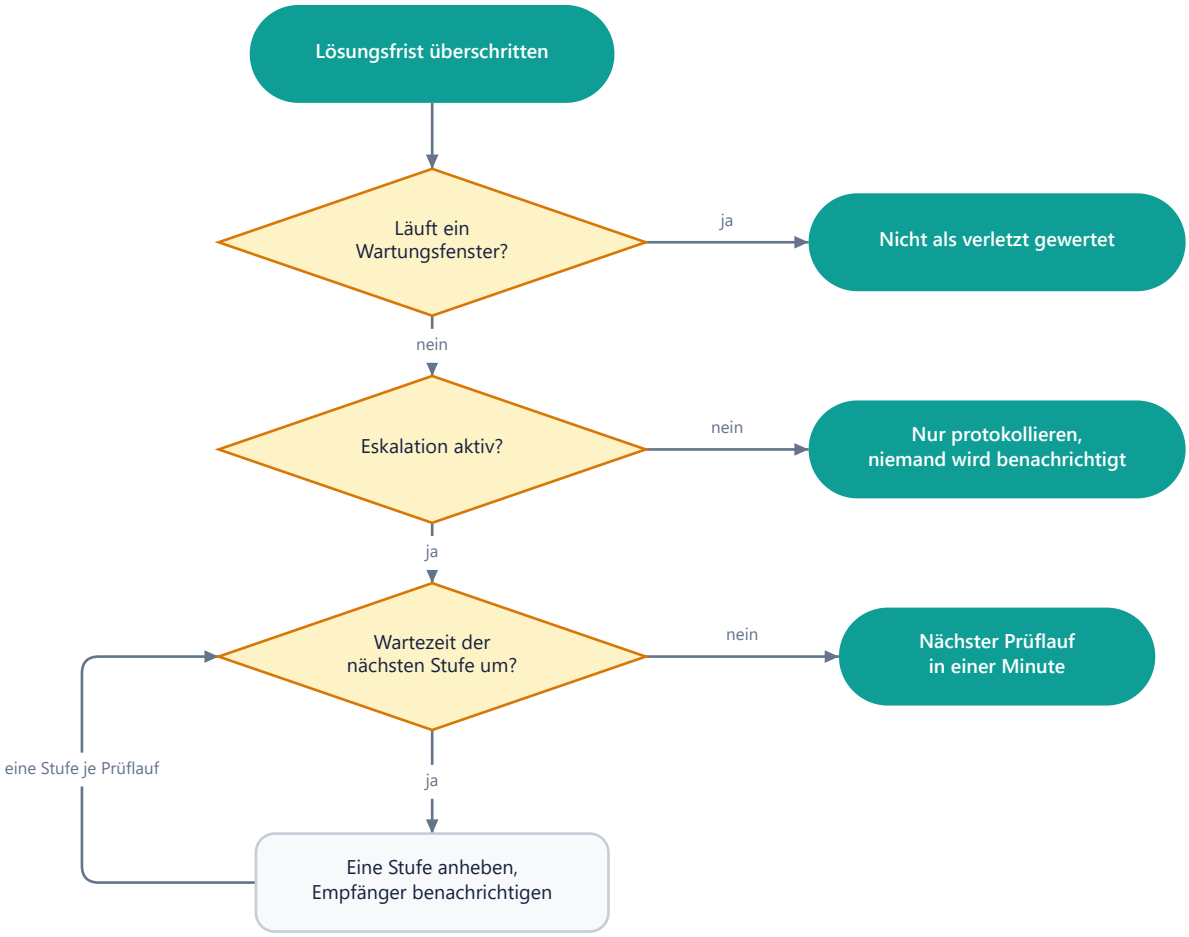

Abb. 4 – Eskalation bei Fristüberschreitung: drei Bedingungen entscheiden, ob überhaupt eine Stufe angehoben wird.

6a Termine am Ticket: Fälligkeit, Zurückstellung, Wiedervorlage

An einem Ticket hängen drei verschiedene Termine, die oft verwechselt werden. Sie beantworten unterschiedliche Fragen und richten sich an unterschiedliche Personen:

Termin	Frage	Wer wird erinnert
Fälligkeit (Lösungsziel)	Bis wann ist es dem Melder zugesagt?	Bearbeiter – die Eskalation zusätzlich die Gruppe
Zurückgestellt bis	Ab wann lohnt es sich, wieder hinzusehen?	Bearbeiter
Wiedervorlage	Wann will *ich persönlich* daran erinnert werden?	Bearbeiter

6a.1 Fälligkeit von Hand setzen

Die Fälligkeit ergibt sich normalerweise aus dem SLA-Profil (Abschnitt 6). In der Seitenleiste des Tickets lässt sie sich mit einem Datums- und Zeitwähler **überschreiben**; die Änderung wird sofort gespeichert.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ein so gesetzter Termin ist als *von Hand gesetzt* vermerkt und verhält sich anders als eine gerechnete Frist: Er **wandert nicht mehr mit**, wenn das Ticket in einem Warte-Status liegt. Bei einer gerechneten Frist verlängert eine Wartezeit das Lösungsziel – sinnvoll, solange niemand eine Zusage gemacht hat. Sobald ein Bearbeiter dem Melder aber einen Termin genannt hat, soll genau dieser Termin stehen bleiben. Die Eskalation an die Gruppe bleibt davon unberührt.

6a.2 Zurückstellen

Wartet ein Ticket auf etwas – eine Lieferung, eine Rückmeldung, ein Wartungsfenster –, setzen Sie es über den Statusdialog in einen Warte-Status. Der Dialog fragt dabei nach **zurückgestellt bis** (Datum **und** Uhrzeit). Verlässt das Ticket den Warte-Status, wird die Zurückstellung automatisch aufgehoben.

Die Zurückstellung gehört zum Ticket, nicht zu einer Person: Sie beschreibt den Vorgang und ist für jeden sichtbar, der das Ticket ansieht.

6a.3 Wiedervorlage

Die **Wiedervorlage** ist die persönliche Merkhilfe des Bearbeiters, unabhängig vom Status. Auch sie trägt Datum und Uhrzeit.

> **Hinweis für Umsteiger von älteren Ständen.** Bis Version 2026.07.28.1912 gab es nur die Wiedervorlage, und sie ließ sich in der Praxis nicht festhalten: Der Erinnerungs-Job löschte den Termin beim Auslösen, und da der Wähler nur ein Datum kannte (also Mitternacht), lag ein auf „heute" gesetzter Termin bereits in der Vergangenheit und verschwand binnen einer Stunde. Beides ist behoben – Termine bleiben nach dem Erinnern stehen, und beide Termine haben eine Uhrzeit.

6a.4 Erinnerungen

Für **alle drei** Termine gilt dieselbe zweistufige Erinnerung an den Bearbeiter:

1. **Vorwarnung** 24 Stunden vor dem Termin,
2. **Meldung bei Fälligkeit**.

Jede Stufe ist ein eigener Ereignistyp im Benachrichtigungskatalog und lässt sich einzeln abschalten (**Einstellungen ▸ Benachrichtigungen**). Der Termin bleibt nach dem Erinnern sichtbar, statt gelöscht zu werden.

6a.5 Termine in der Ticketliste und in „Meine Termine"

In der **Ticketliste** haben Zurückstellung und Wiedervorlage eigene Spalten. Eingefärbt wird nach dem jeweils nächsten Termin: **rot** für abgelaufen, **gelb** für innerhalb der nächsten 24 Stunden. Die Sicht **Nur Termine** blendet alles aus, woran kein Termin hängt. Im Ticket selbst fasst ein Hinweisstreifen oben beide Termine zusammen.

Die persönliche Tagesübersicht heißt **Meine Termine** (früher „Mein Tag") und zeigt alle drei Terminarten nebeneinander. Eine Besonderheit bei der Fälligkeit: Sie erscheint dort **nur, wenn sie im Blickfeld liegt** – überfällig oder innerhalb der nächsten 24 Stunden. Andernfalls stünde dort jedes offene Ticket mit einer SLA-Frist, und die Übersicht wäre wertlos.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

7 Support-Gruppen & Sichtbarkeit

Tickets werden **Support-Gruppen** zugeordnet. Mitglieder einer Gruppe können Benutzer **oder** Rollen sein (`SupportGroupMember`). Agenten sehen nur Tickets ihrer Gruppen – außer mit `tickets.read.all`.

Die Verwaltung erreichen Sie über **Ticketliste** → **Verwaltung** → **Support-Gruppen**. Die Seite ist zweigeteilt: oben die Gruppenliste, darunter die Mitglieder der dort **angeklickten** Gruppe.

Der Mitgliederbereich unterscheidet zwei Dinge, die leicht verwechselt werden:

- **Wer ist in dieser Gruppe** – die tatsächlichen Personen. Eine Rollen-Mitgliedschaft ist eine *Regel*, keine Person; sie wird hier zu den Trägern der Rolle aufgelöst. Zu jedem Namen steht die Begründung („direkt" bzw. „über Rolle X"). Abgelaufene Rollenzuweisungen zählen nicht mit.
- **Mitgliedschafts-Regeln** – die eingetragenen Benutzer und Rollen. Nur diese lassen sich entfernen: Wer über eine Rolle in der Gruppe ist, verschwindet erst, wenn die Rollen-Regel entfernt oder ihm die Rolle entzogen wird.

> **Hinweis:** Eine Gruppe kann „4 Mitglieder" ausweisen und trotzdem niemanden enthalten – nämlich dann, wenn alle vier Einträge Rollen sind, die derzeit keine Träger haben. Genau dafür gibt es die obere Liste.

Das Ändern der Mitglieder verlangt sowohl `ticketing.config.manage` als auch `tickets.assign` – fehlt eines davon, wird der Fehlschlag ausdrücklich gemeldet.

8 Auto-Routing

Regeln weisen eingehende Tickets automatisch zu (z. B. Kategorie „Netzwerk“ → Gruppe „NOC“). Regeln bestehen aus **Bedingung** → **Aktion** (Zuweisung an Gruppe/Agent, Priorität setzen, taggen).

8.1 Kategorie aus dem Betreff setzen

Als **Aktion** kann eine Regel auch die **Kategorie** setzen. Damit kategorisieren Sie Meldungen aus dem Postfach automatisch über **Schlagwörter im Betreff** – z. B. Bedingung *„Betreff enthält Drucker“* → Aktion *Hauptkategorie „Hardware“, Subkategorie „Drucker“*.

- Eine **Subkategorie** wirkt nur zusammen mit der Hauptkategorie **derselben** Regel. Eine Unterkategorie unter einer fremden Hauptkategorie wäre eine erfundene Kombination.
- Hat der Melder **selbst** eine Kategorie angegeben, bleibt diese stehen. Eine Eingabe von Hand ist eine Aussage und wird nicht von einer Regel überschrieben.
- Treffen mehrere Regeln zu, gewinnt die **letzte** – wie bei Priorität und Gruppe.

> **Der praktische Nutzen liegt nicht in der Ordnung, sondern in der Zuständigkeit.** Auf der Kategorie sitzt das Routing zur Support-Gruppe (Abschnitt 8). Setzt eine Regel die Kategorie, wird das Routing anschließend **erneut** ausgewertet – das Ticket landet also nicht nur einsortiert, sondern auch bei der richtigen Gruppe. Bei einer E-Mail ist das der Unterschied zwischen einem Vorgang, der wartet, und einem, der bearbeitet wird.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

9 Genehmigungen & Finanzstufen (Service Request)

Service Requests können **Genehmigungsschritte** erfordern. **Freigabestufen** koppeln Genehmiger an finanzielle Schwellen (z. B. bis 500 € Teamleitung, darüber Bereichsleitung). Erst nach Freigabe wird die Leistung ausgeführt.

10 Problem-Management & KEDB

Wiederkehrende Incidents münden in ein **Problem**. Nach der Ursachenanalyse (RCA) wird ein **Known Error** veröffentlicht – mit Workaround. Verknüpfte Incidents profitieren sofort.

11 Change-Management & CAB

Changes werden **risikobewertet** und über das **Change Advisory Board (CAB)** freigegeben. Ein Change durchläuft die Status **Entwurf** → **Eingereicht** → **Genehmigt** → **In Umsetzung** → **Abgeschlossen** (bzw. *Abgelehnt* / *Zurückgezogen*).

Change-Arten:

- **Normal** – durchläuft die reguläre Genehmigung. Vorgabe für neue Anträge.
- **Standard** – ein vorautorisierter, wiederkehrender Change; er gilt **sofort als genehmigt**.
- **Notfall** – der beschleunigte Weg; die Prüfung erfolgt nachgelagert (ECAB).

11.1 Einen RFC anlegen und einreichen

> 💻 **Client:** *Service Desk* ▸ *Changes* ▸ *Neuer RFC* · 🌐 **Portal:** *Service Desk* ▸ *Changes* · 🔑 `change.read` zum Lesen, `change.manage` zum Anlegen, `change.approve` zum Freigeben

🔒 **Pflichtreihenfolge** — solange ein RFC im **Entwurf** ist, lässt er sich ändern. **Nach dem Einreichen ist er unveränderlich** und Teil der nachvollziehbaren CAB-Historie. Erfassen Sie deshalb alles vollständig, bevor Sie einreichen – ein Nachtrag ist danach nur noch als neuer RFC möglich.

1. Öffnen Sie *Changes* und klicken Sie auf *Neuer RFC*.
2. Erfassen Sie *Titel*, *Beschreibung* und *Begründung*.
3. Wählen Sie die **Art** (*Normal*, *Standard* oder *Notfall*) und die **Risikoklasse**.
4. Setzen Sie **geplanten Start** und **geplantes Ende** – daraus entsteht der Eintrag im Change-Kalender, den alle anderen sehen.
5. Hinterlegen Sie den **Rollback-Plan**: was zu tun ist, wenn die Änderung nicht wirkt.
6. Verknüpfen Sie über das Suchfeld die auslösenden *Tickets* oder *Probleme* und die betroffenen *Configuration Items*.
7. Prüfen Sie alles noch einmal und reichen Sie den RFC ein.

✅ **Ergebnis:** Der Change wechselt von *Entwurf* auf *Eingereicht* und wartet auf die CAB-Entscheidung. Bei der Art *Standard* gilt er sofort als genehmigt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Achtung:** Ein RFC ohne Rollback-Plan ist kein vollständiger Antrag. Die Frage „Und wenn es schiefgeht?" wird im CAB gestellt – besser, sie steht schon im Antrag.

Verknüpfungen: In der Detailansicht verknüpfen Sie einen Change über ein Suchfeld mit den auslösenden **Tickets** oder **Problemen** und den betroffenen **Configuration Items** (CIs) – ohne IDs einzutippen. Das *primäre* betroffene CI bleibt gesondert geführt (für Auswirkungsanalyse und SLA).

> **Hinweis:** Wer den Change **nur sehen** möchte – etwa um zu wissen, wann gewartet wird –, braucht dafür nicht die Change-Liste: Der **Change-Kalender** zeigt dasselbe in zeitlicher Reihenfolge und steht in beiden Oberflächen (ID-10, Abschnitt 12).

Vier-Augen-Prinzip & Ein-Personen-Betrieb: Standardmäßig darf der Antragsteller seinen eigenen Change **nicht** selbst freigeben (Funktionstrennung, ISO 27001 A.5.3). In sehr kleinen Teams lässt sich das über die **Genehmigungs-Einstellungen** (Zahnrad in der Change-Liste) anpassen: Ist der **Ein-Personen-Betrieb** aktiv, darf dieselbe Person Antrag und Freigabe in Personalunion verantworten. Wer grundsätzlich freigeben darf, steuert das Recht *Changes genehmigen*.

12 Kundenzufriedenheit (CSAT)

Nach Lösung kann der Melder eine **Zufriedenheitsbewertung** abgeben. Die Auswertung erscheint in den Service-Desk-Dashboards (ID-27).

12a Arbeiten in der Ticketliste

Neben **„+ Neues Ticket"** stehen die Werkzeuge für das tägliche Arbeiten: Suche, Filter, Export, Bericht, Spaltenwahl, gespeicherte Ansichten und die Legende. Die Verwaltungsobjekte des Service Desks sind im Menü **Verwaltung** zusammengefasst: Auto-Routing-Regeln, Support-Gruppen, Wartungsfenster, Freigabestufen, Textbausteine, Wissensdatenbank und Reporting. Jeder Eintrag erscheint nur, wenn Sie das zugehörige Recht besitzen; die Wissensdatenbank steht allen offen.

Die **Datenqualität** des Service Desks finden Sie unter **Service Desk ▸ Dashboard** – sie ist eine Auswertung und keine Aktion auf der Ticketliste.

Gespeicherte Ansichten halten eine Filterkombination unter einem Namen fest. Die zuletzt gewählte Ansicht gilt als **Vorgabe** und wird beim nächsten Öffnen der Liste automatisch wieder angewandt – bis Sie eine andere wählen oder die aktive löschen. Ein Sprung aus dem Dashboard (etwa auf „offene Tickets") hat Vorrang und übergeht die Vorgabe für diesen Aufruf.

12b Bilder und Dateien ins Journal einfügen

Im Journal-Eingabefeld können Sie Screenshots und Dateien (Bilder, PDF, Textdateien) direkt mit **Strg+V** einfügen oder per Drag & Drop ablegen. Sie werden als **Anhang** des Tickets hochgeladen, erscheinen in

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

der Anhangsliste und werden im Eingabetext als `[Anhang: …]` vermerkt. Reiner Text wird unverändert eingefügt.

13 Konfiguration (Administration)

Alle folgenden Einstellungen finden Sie unter **Einstellungen ▸ Service Desk**:

1. **SLA-Profile** anlegen/bearbeiten (Regeln je Priorität, Standardprofil festlegen).
2. **Geschäftszeiten/Kalender** definieren (Arbeitszeiten, Feiertage).
3. **Kategorien-Baum** pflegen und mit **CMDB-Vorfilter** verknüpfen.
4. **Auto-Routing-Regeln** erstellen.
5. **Freigabestufen** (finanziell) konfigurieren.
6. **Textbausteine** und **Ticket-Vorlagen** verwalten.
7. **Wartungsfenster** planen.

> **Tipp:** Beginnen Sie mit wenigen, klaren Kategorien. Zu feingliedrige Kategorien erschweren die Auswahl für Melder und die Auswertung.
> **Mitgelieferte Einsatzfelder:** Der Standard-Kategorienbaum umfasst neben den IT-Kategorien auch **Vertrieb**, **Gebäudemanagement** und **Personalabteilung** samt Unterkategorien – so lässt sich der Service Desk auch für nicht-IT-Anliegen nutzen. Bestehende Mandanten erhalten fehlende Einsatzfelder automatisch ergänzt, ohne vorhandene Kategorien zu verändern.

14 Rollen & Rechte (Auszug)

Recht	Erlaubt
`tickets.read`	eigene/gruppeneigene Tickets sehen
`tickets.read.all`	alle Tickets sehen
`tickets.create` / `tickets.update`	Tickets anlegen und bearbeiten
`tickets.assign` / `tickets.close` / `tickets.escalate`	Zuweisen, schließen, eskalieren
`ticketing.config.manage` / `ticketing.category.manage`	Service-Desk-Konfiguration und Kategorien verwalten
`tickets.sla.manage`	SLA-Profile pflegen

Verwandte Themen

- **ID-11 – Agentenhandbuch – Service Desk** – wie Agenten damit täglich arbeiten
- **ID-25 – Modul Wissensdatenbank** – bekannte Fehler und Lösungsartikel am Ticket
- **ID-31 – Modul Benachrichtigungen & E-Mail** – wer wann per E-Mail informiert wird

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-41 – Modul Identität, Benutzer & Rollen** – Rollen und Rechte für Bearbeitung und Einsicht
- **ID-27 – Modul Reporting & Dashboards** – SLA-Erfüllung und Aufkommen auswerten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · SERVICE DESK

Monitoring-Anbindung

Alarme einer Überwachung als Vorgänge – Entwarnung, Wartungsfenster, Sturmbremse

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-20a Monitoring-Anbindung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Monitoring-Anbindung

Aus Störungsmeldungen einer Überwachung werden Vorgänge — dedupliziert, mit Entwarnung, ohne Fehlalarm im Wartungsfenster.

INHALT

1 Was diese Anbindung leistet — und was nicht
2 Der Weg einer Meldung
3 Eine Quelle einrichten
4 Das Regelwerk
5 Entwarnung und Nachlaufzeit
6 Die Aufnahme technisch
7 Die Alarm-Übersicht
8 Aufbewahrung der Roh-Ereignisse
9 Rechte
Verwandte Themen

1 Was diese Anbindung leistet — und was nicht

Ordivis Platform **überwacht nicht selbst**. Es nimmt entgegen, was CheckMK, PRTG, Zabbix, Nagios oder ein beliebiges anderes Werkzeug meldet, und macht daraus einen nachvollziehbaren Vorgang.

Der Nutzen liegt nicht im Weiterreichen, sondern im **Wegräumen des Rauschens**:

Ohne Anbindung	Mit Anbindung
Fünfzig Mails zu einer defekten Platte	Ein Vorgang mit Zähler
Nachts geöffnet, morgens von Hand geschlossen	Entwarnung löst automatisch
Wartungsfenster erzeugt Fehlalarme	Meldung wird vermerkt, kein Vorgang
Ein Ausfall des Rechenzentrums flutet das Postfach	Ein Sammelvorgang

2 Der Weg einer Meldung

Jede Meldung nimmt denselben Weg, gleich ob sie per Webhook kommt oder aus einem Postfach:

1. **Idempotenz** — Wurde diese Ereignis-Kennung schon verarbeitet? Dann Ende. Eine Wiederholung nach einem Zustellfehler darf keinen zweiten Vorgang erzeugen.
2. **Regelwerk** — Kategorie, Gruppe, Bearbeiter, Vorgangsart, oder „kein Vorgang".
3. **Episode** — Gibt es zu diesem Korrelationsschlüssel schon eine offene Störung? Dann wird angehängt.
4. **Wartungsfenster** — Läuft eines im Geltungsbereich? Dann wird vermerkt, nicht gemeldet.
5. **Sturmbremse** — Über der Schwelle hängt alles an einem Sammelvorgang.
6. **Vorgang** — Anlegen oder kommentieren.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Warum eine Episode und nicht ein Ereignis. Die Episode ist der Zeitraum von der ersten Meldung bis zur Entwarnung. Sie ist der Grund, warum aus fünfzig Meldungen ein Vorgang wird. Der **Korrelationsschlüssel** entscheidet darüber — üblicherweise `host/dienst` . Wird er falsch gebildet, entstehen entweder fünfzig Vorgänge für eine Störung oder einer für fünfzig verschiedene.

3 Eine Quelle einrichten

💻 **Nur im Client:** *Service Desk* ▸ *Monitoring-Anbindung*, Registerkarte *Quellen* · 🔑 `monitoring.source.manage` — Das Portal zeigt die eingehenden Alarme, richtet die Quellen aber nicht ein.

Sehen Sie die Registerkarte *Quellen* nicht, fehlt Ihnen das Recht. Der Menüpunkt selbst steht dem ganzen Service Desk offen (Abschnitt 7); die Quellen und der Knopf *Quelle anlegen* erscheinen nur mit `monitoring.source.manage` . Die Seite öffnet dann gleich auf *Alarme*.

🔒 **Pflichtreihenfolge** — die **Art** der Quelle belegt die Schweregrad-Abbildung vor. Wer sie zuletzt setzt, überschreibt seine eigenen Einstellungen.

1. Öffnen Sie *Service Desk* ▸ *Monitoring-Anbindung* und wechseln Sie auf die Registerkarte *Quellen*.
2. Klicken Sie auf *Quelle anlegen*.
3. Wählen Sie die **Art** des Überwachungswerkzeugs.
4. Prüfen Sie die vorbelegte **Schweregrad-Abbildung** und passen Sie sie an Ihre Skala an.
5. Stellen Sie **Nachlaufzeit**, **Sturmschwelle** und **Flatter-Erkennung** ein.
6. Hinterlegen Sie den **Tieflink** mit den Platzhaltern `{host}` und `{service}`, damit ein Klick im Vorgang zum Sensor springt.
7. Speichern Sie.

✅ **Ergebnis:** Die Quelle nimmt Meldungen entgegen. Ihr letzter Empfang steht ab jetzt in der Alarm-Übersicht (Abschnitt 7) – der Wert, an dem Sie einen stillen Ausfall erkennen.

Je Quelle einstellbar — und das ist Absicht: Wer CheckMK und PRTG nebeneinander betreibt, hat zwei Schweregrad-Skalen und zwei Meldefrequenzen.

Einstellung	Bedeutung	Vorgabe
Art	Bestimmt die vorbelegte Schweregrad-Abbildung	—
Schweregrad-Abbildung	`{"crit":2,"warn":1}` — welcher Schweregrad welche Priorität ergibt	Vorbelegung des Werkzeugs
Nachlaufzeit	Minuten zwischen Entwarnung und Lösen	10
Sturmschwelle	Meldungen je Minute, ab der gesammelt wird	20
Flatter-Erkennung	Zustandswechsel in einem Zeitfenster	5 in 30 min

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Einstellung	Bedeutung	Vorgabe
Vorrang menschlicher Arbeit	Entwarnung kommentiert nur, wenn jemand am Vorgang war	an
Tieflink	Muster mit `{host}` und `{service}` — Sprung zum Sensor	—

Ein unbekannter Schweregrad fällt auf die Vorgabe, nicht durch. Eine Meldung zu verwerfen, weil ihre Stufe nicht in der Tabelle steht, ist der stille Ausfall, den niemand bemerkt: Die Überwachung meldet, Ordivis Platform schweigt, und beide halten sich für zuständig.

4 Das Regelwerk

Bedingungen (alle gesetzten UND-verknüpft): Host-Muster, Dienst-Muster, Schweregrad, Label. Platzhalter ist `*` — bewusst kein regulärer Ausdruck, damit ein versehentlich gültiger Ausdruck keine unerwartete Menge trifft. Eine Regel **ohne** Bedingung trifft auf alles zu; so baut man eine Auffangregel ans Ende.

Aktionen: Kategorie, Support-Gruppe, Bearbeiter, Vorgangsart, Priorität — oder **kein Vorgang**.

Die Auswertung folgt derselben Semantik wie die Ticket-Automatisierungsregeln: Reihenfolge über die Sortierung, spätere Regeln überschreiben frühere, `Kette beenden` bricht ab.

Unterdrückung lässt sich nicht zurücknehmen. Hat eine Regel entschieden, dass kein Vorgang entsteht, kann eine spätere das nicht aufheben. Sonst hinge die Wirkung an der Sortierreihenfolge, und eine Ausnahme („dieser Testserver nie") wäre nicht verlässlich. Der Alarm wird trotzdem geführt und ist in der Alarm-Übersicht sichtbar — **unterdrückt heißt nicht verschluckt**.

5 Entwarnung und Nachlaufzeit

Meldet die Überwachung Entwarnung, wird der Vorgang **nicht sofort** gelöst: Die Nachlaufzeit läuft, und ein erneutes Problem in diesem Fenster hebt sie auf. Ohne sie öffnete und schlösse eine flatternde Prüfung im Minutentakt Vorgänge.

Vorrang menschlicher Arbeit: Wurde am Vorgang gearbeitet — der Status ist nicht mehr „Neu", oder es gibt einen Eintrag eines Menschen im Verlauf —, kommentiert die Entwarnung nur. Eine Überwachung weiß nicht, ob jemand noch prüft.

6 Die Aufnahme technisch

`POST /api/v1/monitoring/events` · Recht `monitoring.event.ingest` · **Stapel** mit Quittung je Ereignis

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
{
  "schemaVersion": 1,
  "sourceId": "…",
  "events": [
    {
      "eventId": "checkmk-4711",
      "host": "srv-db-01",
      "service": "Speicherplatz /var",
      "severity": "crit",
      "eventType": "problem",
      "subject": "srv-db-01: /var zu 95 % belegt",
      "message": "FILESYSTEM /var 95.2% used",
      "labels": ["umgebung=produktiv"]
    }
  ]
}
```

`eventType` ist `problem` oder `recovery`. Die Antwort quittiert **je Ereignis** — ein Stapel darf nicht an einer krummen Zeile scheitern.

> **Die Ratelimit-Richtlinie ist eine eigene** (600 Anfragen je Minute), nicht die allgemeine mit 30. Im Sturmfall — also genau dann, wenn die Meldungen zählen — liefe die Überwachung sonst in Abweisungen und verwürfe ihre Alarme. Der Sturm wird **fachlich** gebremst, nicht dadurch, dass die Leitung zumacht.

7 Die Alarm-Übersicht

> 💻 **Client:** *Service Desk* ▸ *Monitoring-Anbindung*, Registerkarten *Alarme* und *Wiederkehrend* · 🔑 `tickets.read`

Die Registerkarte *Alarme* zeigt offene, entwarnte und **unterdrückte** Störungen mit dem Grund der Unterdrückung. *Wiederkehrend* nennt die Korrelationsschlüssel, die in den letzten 30 Tagen mehrfach aufgetreten sind — die Kandidaten für das Problem-Management.

> **Warum diese Seite dem Service Desk offensteht und nicht nur der Administration.** Ein **unterdrückter** Alarm hat per Definition keinen Vorgang; in einer Vorgangsliste kann von ihm nichts stehen. Die Zusage aus Abschnitt 4 — „unterdrückt heißt nicht verschluckt" — lässt sich also nur hier nachprüfen. Wäre die Seite der Administration vorbehalten, könnte eine zu weit geratene Unterdrückungsregel außerhalb dieses Kreises niemandem auffallen.

So prüfen Sie turnusmäßig, ob die Anbindung noch liefert:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

💻 **Client:** *Service Desk* ▸ *Monitoring-Anbindung*, Registerkarte *Quellen* · 🔑 `monitoring.source.manage`

1. Öffnen Sie *Service Desk* ▸ *Monitoring-Anbindung* und bleiben Sie auf der Registerkarte *Quellen*.
2. Lesen Sie je Quelle die Zahl der offenen Alarme und den **letzten Empfang** ab.
3. Vergleichen Sie den Wert mit der erwarteten Meldefrequenz der Quelle.

✅ **Ergebnis:** Sie erkennen eine tote Anbindung, bevor sie im Ernstfall auffällt.

Der letzte Empfang ist der wichtigste Wert der Seite. Eine Anbindung, die seit drei Tagen nichts mehr liefert, sieht in jeder anderen Ansicht aus wie **Ruhe** — und ist in Wahrheit ein stiller Ausfall. ⚠️ Diese Kontrolle hängt an `monitoring.source.manage`: Der letzte Empfang steht bei der Quelle, und die Quellen sieht nur, wer sie verwalten darf. Wer die Überwachung der Überwachung breiter aufstellen will, muss dieses Recht vergeben — eine Ersatzansicht dafür gibt es nicht.

8 Aufbewahrung der Roh-Ereignisse

Jede eingehende Meldung wird im Wortlaut abgelegt — sie ist der Beleg, wenn der Betreiber der Überwachung und Ihr Haus sich über eine Störung uneinig sind. Diese Rohdaten wachsen aber mit der **Ereignismenge**, nicht mit der Vorgangsmenge: Eine Überwachung mit tausend Prüfungen schreibt im Sturmfall Zehntausende Zeilen am Tag.

Ein nächtlicher Lauf kürzt sie deshalb:

Was	Roh-Ereignisse der Überwachung
Aufbewahrung	**30 Tage**
Wann	täglich um 03:15 UTC, je Mandant
Auftrag	*Monitoring-Rohereignisse kürzen* (`ticketing.monitoring-event-retention`)

Was bleiben bleibt. Die **Episoden** — also die Geschichte der Störungen mit Beginn, Entwarnung und Vorgangsbezug — werden **nicht** gekürzt. Und Ereignisse einer Episode, die noch **offen** ist oder in der Nachlaufzeit steht, bleiben ebenfalls stehen, auch wenn sie älter sind: Den Wortlaut ausgerechnet für einen noch laufenden Fall zu löschen, nähme Ihnen den Beleg im laufenden Streit. Eine flatternde Platte, die seit Wochen offen steht, behält damit ihre gesamte Vorgeschichte.

Der Auftrag steht wie jeder andere unter **Administration** ▸ **Aufträge**; Zeitpunkt und Aktivierung lassen sich dort ändern.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

9 Rechte

Recht	Wer
`monitoring.event.ingest`	Das Integrations-Token der Quelle
`monitoring.source.manage`	Mandanten-Administration — die Registerkarte *Quellen*, der letzte Empfang und das Anlegen
`tickets.read`	Der Service Desk — Alarme, wiederkehrende Störungen, Kennzahlen, und damit der Menüpunkt selbst

Bewusst zwei Rechte: Wer eine Überwachung anbinden darf, muss nicht auch Störungen einspeisen dürfen — und ein Token soll nichts konfigurieren können.

Und bewusst **eine Seite für beide**: Die Alarme sind Tagesgeschäft des Service Desk, die Quellen sind Einrichtung. Sie zu trennen hieße, den unterdrückten Alarm von der Regel zu trennen, die ihn unterdrückt hat — und genau dieser Weg wird gegangen, wenn jemand fragt, warum zu einer Störung kein Vorgang entstanden ist.

Verwandte Themen

- Service Desk / Ticketing — Wartungsfenster, Automatisierungsregeln
- CMDB — die CI-Zuordnung über den Hostnamen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · CMDB

Modul CMDB (Configuration Management)

Configuration Management – CIs, Klassen, Beziehungen, Topologie, Datenqualität

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-21 CMDB

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul CMDB (Configuration Management)

Stellen Sie sich vor, jemand fragt Sie: „Wenn dieser Server heute Nacht ausfällt – welche Programme, welche Abteilungen, welche Kunden merken das?“ In vielen Organisationen kann das niemand vollständig beantworten. Das Wissen steckt in einzelnen Köpfen, in alten Excel-Listen und in Netzplänen, die seit dem letzten Umzug niemand mehr angefasst hat.

Genau diese Lücke füllt eine CMDB. Die Abkürzung steht für *Configuration Management Database*, also eine Datenbank zur Verwaltung der IT-Konfiguration. Dahinter steckt ein zentrales Verzeichnis aller Bausteine Ihrer IT: Server, Arbeitsplätze, Anwendungen, Netzwerkgeräte, Dienste und Standorte. Jeder einzelne Baustein heißt **Configuration Item** (CI). Wichtig ist dabei nicht die Liste allein, sondern dass die CMDB festhält, wie die Bausteine zusammenhängen: Diese Anwendung läuft auf jenem Server, der Server steht in diesem Rechenzentrum, an diesem Dienst hängt jene Abteilung.

So wird aus einer bloßen Inventarliste ein Werkzeug, mit dem Sie Auswirkungen abschätzen, Änderungen vorbereiten und Störungen schneller eingrenzen. In Ordivis Platform ist die CMDB die Datengrundlage, auf die auch Service Desk, Asset-Management und das Sicherheitsmodul zugreifen.

INHALT

1 Zweck & Nutzen
2 Grundbegriffe
3 CI anlegen & bearbeiten
4 CI-Klassen & Attribute
5 Beziehungen
6 Standortbaum (Drag & Drop)
7 Netzwerkschnittstellen am CI
8 Topologie- & Business-Service-Map
9 Installierte Software & Inventar
10 Anwendungen und Verantwortlichkeiten
11 Discovery-Abgleich
12 Prozessmanagement (Geschäftsprozessdokumentation)
13 Etiketten und NFC-Tags am Gerät
13.1 Was Sie kennzeichnen können
13.2 Aussehen des Etiketts festlegen
13.3 Etikettendrucker und Medien
13.4 Etikett für ein Gerät drucken
13.5 NFC-Tag zuordnen
13.6 Vor Ort scannen
14 Datenqualität
15 Abgrenzung CMDB ↔ Asset Management
16 Rollen & Rechte (Auszug)
Verwandte Themen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1 Zweck & Nutzen

- Vollständiges, aktuelles Bild der IT-Landschaft.
- Auswirkungsanalyse: „Welche Dienste hängen an diesem Server?“
- Grundlage für Change-, Incident- und Problem-Management.

2 Grundbegriffe

Begriff	Bedeutung
CI	Ein dokumentierter IT-Baustein (Server, PC, Dienst, Standort, Anwendung …).
CI-Klasse	Bauplan eines CI-Typs mit definierten **Attributen** (im Backend definiert).
Beziehung	Typisierte Verbindung zwischen CIs (z. B. „läuft auf“, „hängt ab von“).
Standortbaum	Hierarchische Einordnung von CIs nach Standort/Enthaltensein.

3 CI anlegen & bearbeiten

Die zentrale Liste **Configuration Items** zeigt alle CIs mit Name, Status, Klasse, Hersteller und Seriennummer. Über die Kopfzeile filtern Sie nach Kategorie, Klasse und Status, durchsuchen den Bestand und exportieren die Ansicht.

Abbildung 1 zeigt die Liste mit Filtern und Spaltenauswahl.

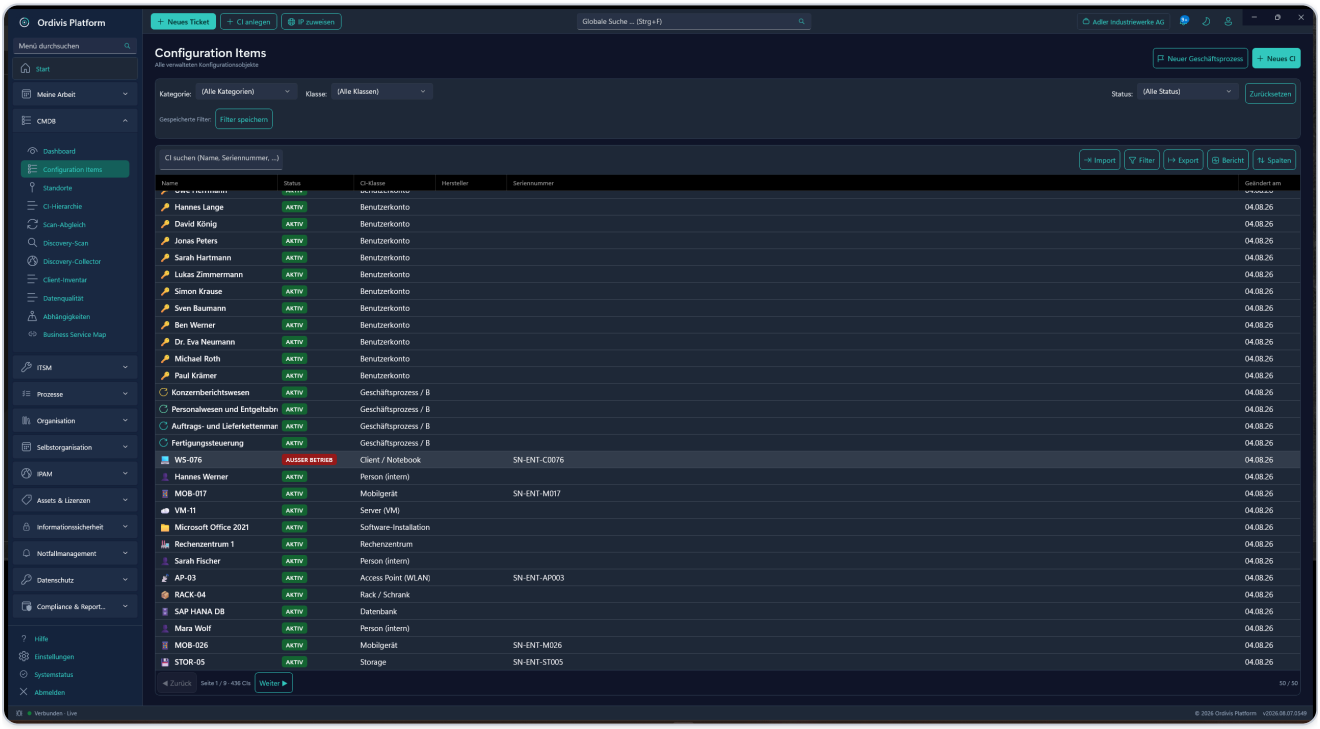

Abb. 1 – Die CI-Liste: Filter (Kategorie/Klasse/Status), Suche, Import/Export und Spaltenauswahl.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> 💻 **Client:** *CMDB* ▸ *Configuration Items* ▸ *Neues CI* · 🌐 **Portal:** *CMDB* ▸ *CMDB-Suche* · 🔑 `cmdb.ci.read` zum Lesen, `cmdb.ci.update` zum Anlegen und Ändern — Das Portal zeigt CIs und ihre Beziehungen; angelegt und umklassifiziert wird im Client.

🔒 **Pflichtreihenfolge** — die **CI-Klasse** bestimmt, welche Attributfelder überhaupt erscheinen. Wird sie erst nachträglich gewechselt, ist ein Teil des bereits Erfassten hinfällig.

1. Öffnen Sie *CMDB* ▸ *Configuration Items* und klicken Sie auf *Neues CI*.
2. Wählen Sie die passende **CI-Klasse** (z. B. „Server“).
3. Füllen Sie die **Attribute** aus. Welche Felder erscheinen, bestimmt die Klasse (jede Klasse bringt eine vollständige, praxisnahe Vorlage mit, siehe Abschnitt 4).
4. **Verweisfelder** (z. B. Verantwortliche/r, Hersteller, Standort, zugewiesene Person) füllen Sie über ein **durchsuchbares CI-Dropdown**: Namen eintippen, Treffer wählen – die Verknüpfung wird als Klarname angezeigt, nicht als technische ID. Das gilt beim Anlegen, in der Detailseite und beim Umklassifizieren.
5. Speichern Sie.

✅ **Ergebnis:** Das CI erscheint in der Liste und im Standortbaum und trägt eine automatisch vergebene CI-Nummer.

Ein Klick auf ein CI öffnet die Detailseite mit Basisdaten, Garantie- und Wartungsangaben, dem Lebenszyklus (Lifecycle) sowie den Reitern für Attribute, Relationen, Schnittstellen und mehr.

Abbildung 2 zeigt eine solche Detailseite am Beispiel eines Servers.

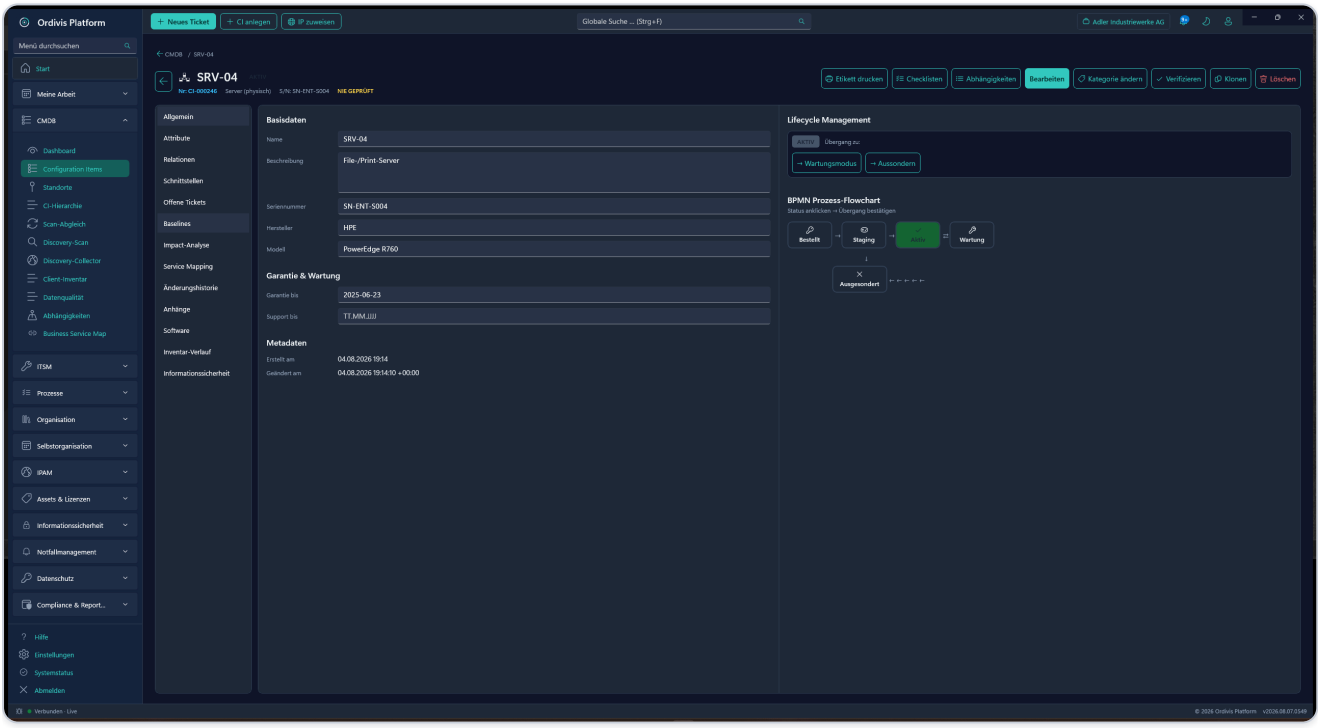

Abb. 2 – CI-Detail (Beispiel Server): Basisdaten, Garantie/Wartung, Lifecycle und BPMN-Statusfluss.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Wichtig: Beim Datum ist das deutsche Format zulässig; ein leeres Seriennummernfeld wird korrekt als „nicht gesetzt“ behandelt (kein Fehler bei mehreren leeren Werten).

Hinweis: Werte aus anderen Modulen (z. B. IP-Adresse aus IPAM) werden **read-only über die Verknüpfung** angezeigt und niemals in CMDB-Felder kopiert.

4 CI-Klassen & Attribute

CI-Klassen und ihre Attribute werden **im Backend** definiert (Seed / CI-Klassen-Verwaltung), nicht im Client hartkodiert. Neue Attribute legen Sie zentral an – sie erscheinen dann automatisch im passenden CI-Formular. Die Klassenverwaltung beschreibt das Administrationshandbuch (ID-40).

Vollständige Vorlagen je Kategorie: Jede mitgelieferte CI-Klasse besitzt einen ausgearbeiteten, an Industriestandards (ITIL SACM, ISO 19770/55000, BSI) orientierten Attributsatz – gruppiert in sinnvolle Abschnitte, mit passenden Feldtypen (Liste, Datum, Ja/Nein, IP/MAC, verschlüsselt, CI-Verweis). Abgedeckt sind u. a. **Server/VM/Client/Mobilgerät/Drucker/Storage**, **Switch/Router/Firewall/Access Point/VLAN**, **Betriebssystem/Datenbank/Applikation/Web-Dienst-API**, **Standort/Gebäude/Etage/Raum/Rack/Rechenzentrum**, **Abteilung/Person (intern & extern)/Hersteller-Lieferant** sowie **Geschäftsprozess** und die Governance-Objekte (ISMS, Information, Dokument, Zertifikat, Schlüssel, Digitale Identität). Eigene Anpassungen (Tailoring) bleiben bei Updates erhalten; fehlende Standard-Felder werden ergänzt, vorhandene nie überschrieben.

5 Beziehungen

Beziehungen nutzen einen **Relationstypen-Katalog** (z. B. „läuft auf“, „verbunden mit“, „Teil von“). Gespeichert wird das Label. So dokumentieren Sie Abhängigkeiten, die für Impact und Service-Map genutzt werden.

6 Standortbaum (Drag & Drop)

Im Standortbaum ordnen Sie CIs hierarchisch ein. Per **Drag & Drop** hängen Sie ein CI unter ein anderes um (Reparent). Die Containment-Richtung ist zentral geregelt.

Abbildung 3 zeigt den Standortbaum mit Gebäuden, Etagen und Racks.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

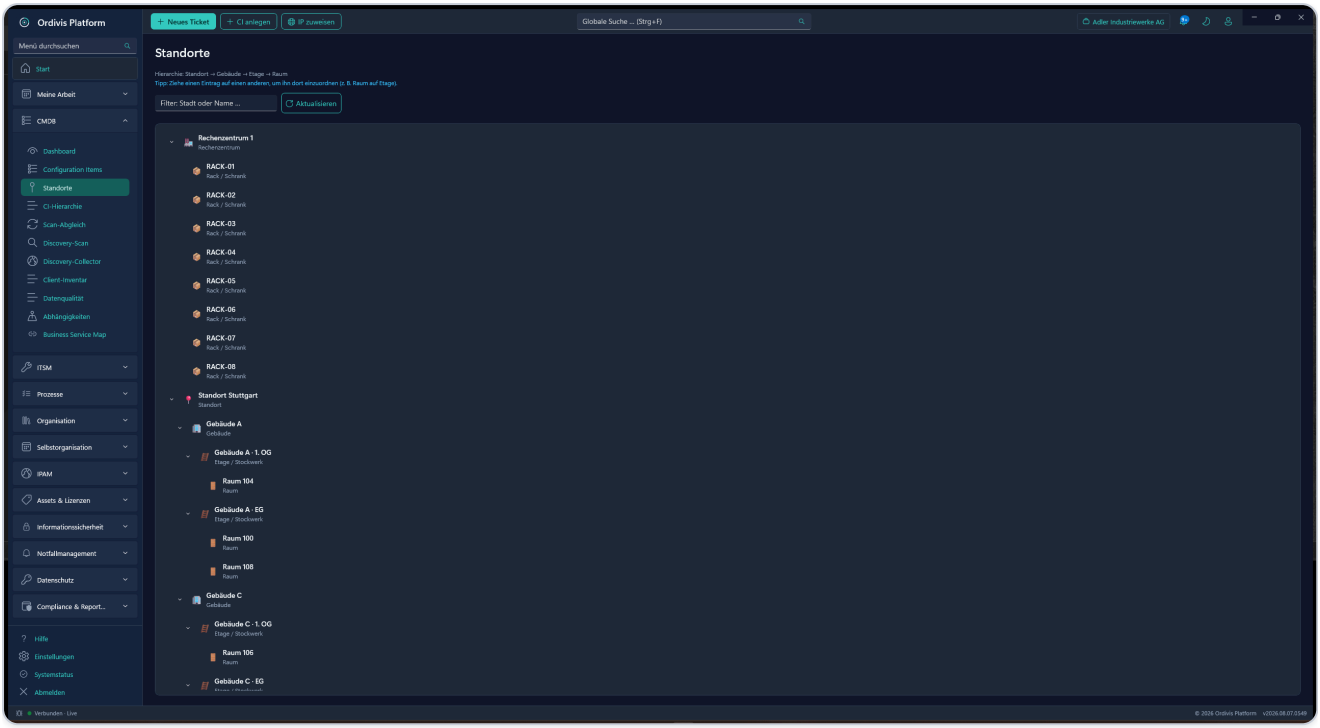

Abb. 3 – Standorte: Gebäude, Etagen und Racks; per Drag & Drop ordnen Sie einen Eintrag einem anderen unter.

7 Netzwerkschnittstellen am CI

Jedes CI kann eigene **Netzwerkschnittstellen** besitzen (eigene Entität, kein „Port = CI“):

- MAC-Adresse, Bezeichnung, Typ.
- **IP-Adresse** wird **read-only aus IPAM** angezeigt.
- **Reziproke Verkabelung:** Verbinden Sie zwei Schnittstellen, wird die Peer-Beziehung beidseitig gepflegt.

Den Bereich finden Sie im CI-Detail unter **Schnittstellen**.

8 Topologie- & Business-Service-Map

- **Topologie-Map:** grafische Sicht der CIs und ihrer Verbindungen (dreispaltige Ansicht mit Zeitleiste).
- **Business-Service-Map:** zeigt links-nach-rechts, welche Dienste von welchen CIs abhängen – ideal für Impact-Analysen.

Die **CI-Hierarchie** stellt denselben Zusammenhang als aufklappbaren Baum dar. Oben wählen Sie den **Relationstyp** (z. B. „läuft auf“), nach dem der Baum gebildet wird – so sehen Sie etwa, welche virtuellen Maschinen auf welchem physischen Server laufen.

Abbildung 4 zeigt einen solchen Baum.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

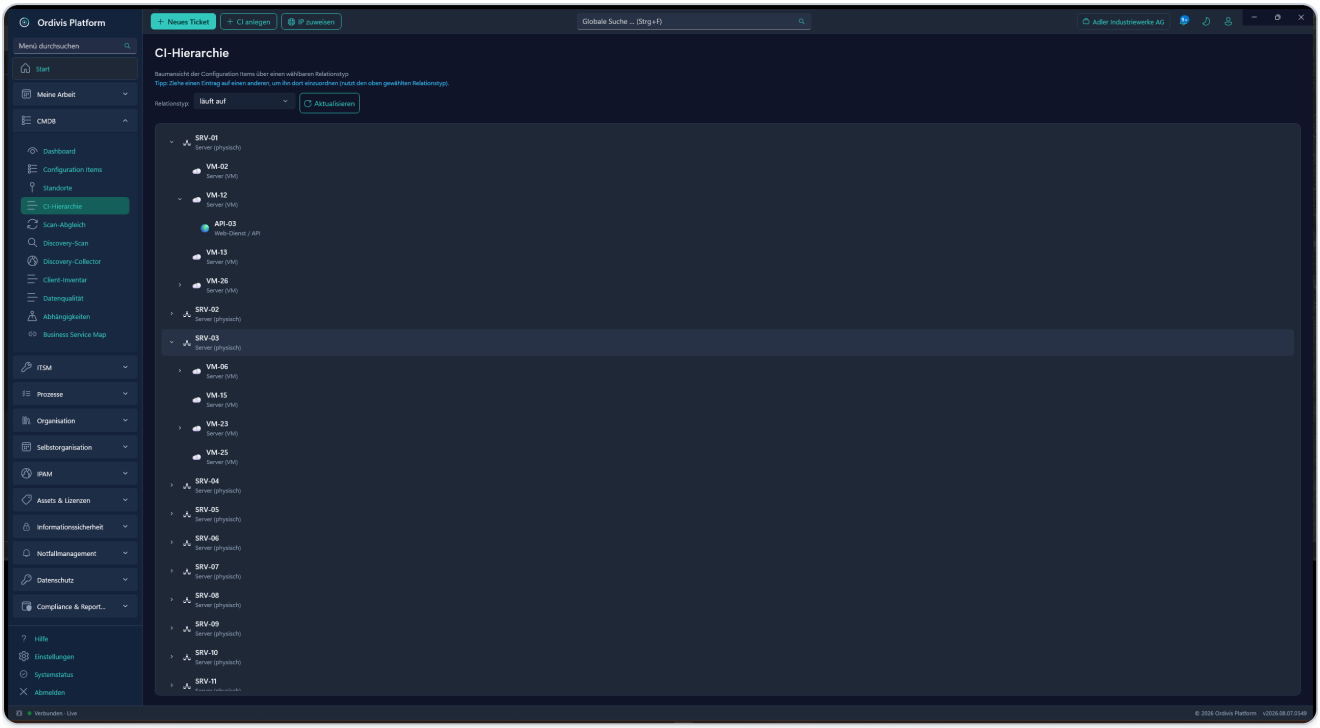

Abb. 4 – CI-Hierarchie: Baum nach Relationstyp „läuft auf" (physische Server mit ihren VMs).

9 Installierte Software & Inventar

Je CI führt ein Tab **Installierte Software** (Hersteller, Name, SKU, Version) – Basis der Lizenzzählung (ID-23). Der **Inventar-Verlauf** speichert versionierte Scan-Snapshots und zeigt den **Diff** der beiden neuesten Aufnahmen. Die Daten stammen aus dem Discovery-Modul (ID-24).

10 Anwendungen und Verantwortlichkeiten

Nicht jede Software ist eine Anwendung. Der **Software-Bestand** zeigt, was installiert ist; die **Anwendungsübersicht** zeigt, was das Haus fachlich führt — mit **Produktbesitzer** und **Vertretung**.

Der Produktbesitzer ist die Person, die die Anwendung am besten kennt und darin Benutzer und Berechtigungen verwaltet. Die IT bleibt in jedem Fall für den technischen Betrieb zuständig; die Anwendungsverantwortung ist eine organisatorische Ergänzung, keine Abgabe.

Zwei Wege führen in die Liste: Im Reiter **Nicht geführt** steht jedes Produkt des Software-Bestands ohne Anwendung — ein Klick auf **Als Anwendung führen** nimmt es auf. Gehostete Fachverfahren ohne lokale Installation legen Sie direkt als Anwendung an.

Alles, was nicht geführt wird, liegt in der Verantwortung der IT. Diese Regel bleibt sichtbar: Der Reiter **Nicht geführt** zählt sie, und in der Übersicht steht bei einer Anwendung ohne Produktbesitzer die Vorgabe **IT-Abteilung** in blasser Schrift statt einer leeren Zelle.

Je Anwendung gibt es höchstens einen Produktbesitzer und höchstens eine Vertretung — sonst entscheidet im Zweifel niemand. Der Reiter **Meine Anwendungen** zeigt jedem Konto, wofür es eingetragen ist.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

11 Discovery-Abgleich

Scan-Treffer erscheinen als **Kandidaten** im **Scan-Abgleich**. Erst beim **Übernehmen** entstehen oder aktualisieren sich CIs (inkl. Software-Installationen und Inventar-Snapshot). So behalten Sie die Kontrolle über automatisch erkannte Objekte.

12 Prozessmanagement (Geschäftsprozessdokumentation)

Das **Prozessmanagement** ist ein eigener Bereich („Prozesse") und orientiert sich an den etablierten Rahmenwerken (GPM/Geschäftsprozessmanagement, **Lean**/KVP, **ISO 9001:2015**, IKS). Ein Geschäftsprozess ist ein CI der Klasse „Geschäftsprozess" – die Prozessdokumentation ist ein **Overlay** darauf, nichts wird dupliziert.

Der Bereich **„Prozesse"** bündelt zwei Sichten: die **Prozessübersicht** – eine tabellarische Gesamtliste aller Prozesse mit ihren Steckbrief-Feldern, durchsuchbar, filterbar, sortierbar und als CSV/PDF exportierbar (ein Doppelklick öffnet den Prozess in der Dokumentation) – und die **Prozessdokumentation** für die Detailsicht eines einzelnen Prozesses. Die Detailsicht ist in **drei Registerkarten** gegliedert:

1. Steckbrief – die zentrale Übersicht mit *allen* steuerungsrelevanten Metadaten auf einen Blick: Zweck/Fachaufgabe, Ziel (Mehrwert), Auslöser → Ergebnis, Eingaben/Ausgaben, Business Capability, Organisationseinheit und Detaillierungsgrad. Dazu erfasst der Steckbrief den **rechtlich-fachlichen Rahmen** (Rechtsgrundlage, Geltungsbereich, Version und Gültigkeit), den **Leistungsempfänger**, die **Entscheidungskriterien**, **Ausnahmen/Fehlerpfade**, **Medienbrüche** sowie **gesetzliche Fristen** und **Gebühren**. Der Steckbrief lässt sich direkt über **Bearbeiten** pflegen.

Die Verantwortung wird **ausschließlich** in der **RACI-Matrix** geführt – *Accountable* (Prozessverantwortliche/r) und *Responsible* (Prozessmanager/in) sowie Consulted/Informed; Personen werden über ein **durchsuchbares Dropdown** der Mandanten-Benutzer ausgewählt.

> **Geändert in 2026.07.28.1912.** Bis dahin standen „Verantwortlich (A)" und „Manager (R)" **zusätzlich** als Zeilen im Steckbrief – zwei gleichrangig pflegbare Stellen für dieselbe Aussage, die in der Praxis auseinanderliefen, ohne dass die Oberfläche sagte, welche gilt. Die Steckbrief-Zeilen entfallen; die Matrix ist die Quelle. Vorhandene Angaben aus dem Steckbrief werden bei der Aktualisierung **einmalig in die Matrix übernommen**, bereits vorhandene Matrix-Einträge bleiben dabei unangetastet. Übersicht und Prüfungsliste lesen die Namen seitdem ebenfalls aus der Matrix.

PDCA-Zyklus (KVP): Der Steckbrief zeigt die aktuelle **PDCA-Phase** (Plan · Do · Check · Act), das **Überprüfungsintervall** sowie die letzte und nächste Überprüfung. Aus letzter Überprüfung + Intervall wird die nächste Fälligkeit fortgeschrieben; eine **Wiedervorlage/Erinnerung** geht rechtzeitig an den/die **Prozessverantwortliche/n** (RACI-Accountable) – so veraltet die Dokumentation nicht still.

2. Prozessmodell (BPMN) – ein **eingebauter grafischer BPMN-2.0-Editor** (bpmn-js Modeler, offline im WebView2, kein CDN): Elemente per **Drag & Drop** aus der Palette, **deutsche** Oberfläche, MD3-Design und eine **Standardvorlage mit Swimlanes** („Eingang / Verarbeitung / Ergebnis"). Ein **Eigenschaften-Panel** bringt den **Namens-Linter** (Objekt+Verb bei Aufgaben, Partizip bei Ereignissen, Frage bei Verzweigungen) direkt ans Element und markiert **IKS-Kontroll-/Risikopunkte**. Speichern erzeugt eine **neue Entwurfs-**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Version; freigegebene Stände sind **schreibgeschützt** (Vier-Augen-Freigabe). Wer extern zeichnet, kann weiterhin **BPMN/XML importieren**. Der Editor steht auch im **Web-Portal** (Rolle Process Editor, SVG-/PNG-Export) bereit.

3. Dokumente – mitgeltende Unterlagen (Richtlinien, Arbeitsanweisungen, Formulare) je Prozess: hinzufügen, herunterladen, **anzeigen** (PDF öffnet im eigenen Viewer-Fenster) und löschen.

Ergänzend: **KPIs** (Ziel/Ist, ISO-9001-Leistungsindikatoren), **IKS-Kontrollmatrix** und **Soll/Ist-Abgleich** gegen ein importiertes Ereignisprotokoll (Process-Mining-Andockung).

Öffentliche Verwaltung: Ist ein Mandant über die Einstellung **Organisationsart** (Einstellungen → Module, je Mandant durch den SuperAdmin) als **„Öffentliche Verwaltung"** gekennzeichnet, blendet der Steckbrief zusätzliche Verwaltungsfelder ein: **LeiKa-ID**, **FIM-ID**, **KGSt-Produktnummer**, **OZG-Themenfeld** und -**Leistungs-ID**, **Reifegrad** (E-Government), **Vertrauensniveau** (eIDAS: Basis/Substantiell/Hoch), **Föderalismus-Ebene** und **Verwaltungsakt** (§ 35 VwVfG). In der Prozessübersicht erscheinen diese Angaben als zusätzliche Spalten. Für alle anderen Mandanten bleiben die Felder ausgeblendet.

Abbildung 5 zeigt den Steckbrief und die Registerkarten daneben.

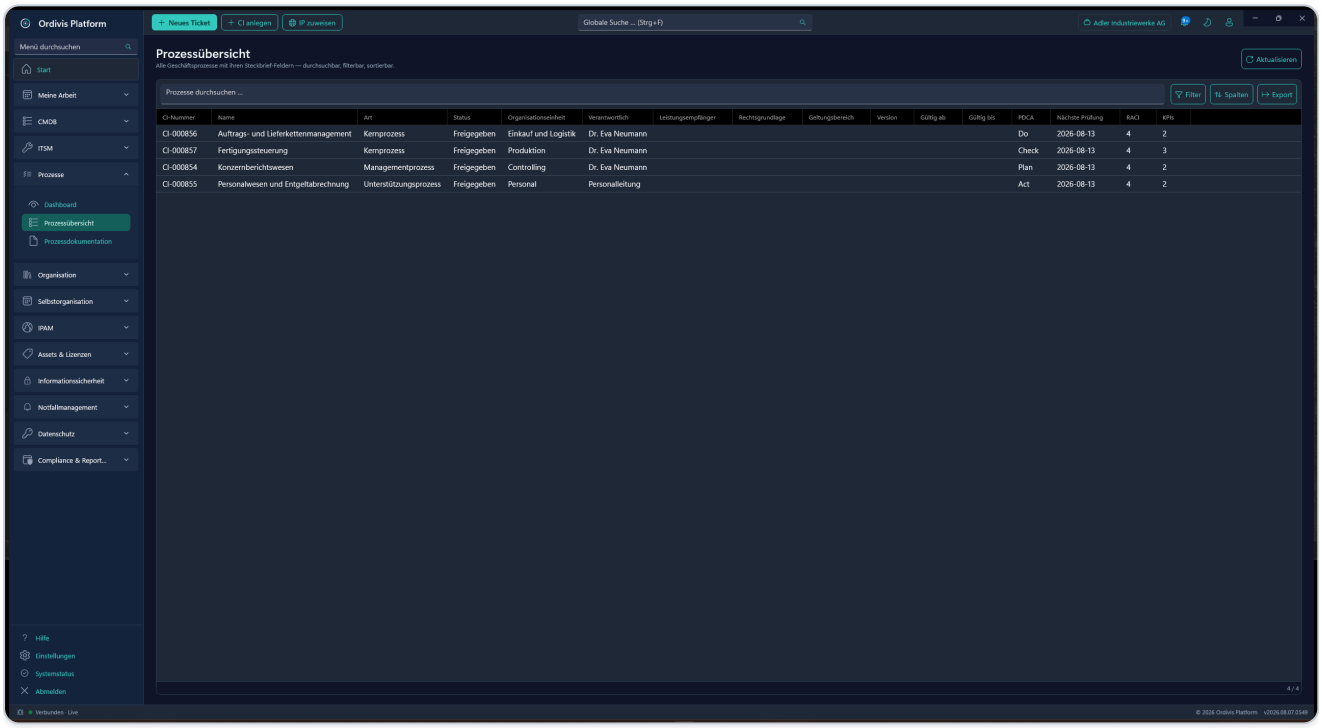

Abb. 5 – Prozessmanagement: Steckbrief (zentrale Übersicht) mit RACI und PDCA-Zyklus, daneben die Registerkarten Prozessmodell (BPMN-Editor) und Dokumente.

13 Etiketten und NFC-Tags am Gerät

Ein Gerät im Regal sagt Ihnen nicht, welcher Datensatz zu ihm gehört. Damit die Zuordnung auch vor Ort gelingt, kennt Ordivis Platform zwei Kennzeichen – und Sie können **beide gleichzeitig** verwenden.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

13.1 Was Sie kennzeichnen können

Kennzeichen	Wie es gelesen wird	Stärke	Grenze
Aufkleber mit QR-Code, DataMatrix oder Strichcode	Handy-Kamera oder Handscanner	Auf Metall und in Kälte zuverlässig; kostet nichts außer dem Etikett	Muss sichtbar und sauber sein – hinter dem Gerät, im geschlossenen Schrank oder unter Staub nicht lesbar
NFC-Tag (Aufkleber oder Anhänger)	Handy an das Gerät halten	Wird durch Kunststoff und im Dunkeln gelesen; kein Blickkontakt nötig	Auf Metall nur mit Ferrit-Untergrund; braucht ein Telefon mit NFC

Die beiden Schwächen liegen genau umgekehrt. Wer an schwer zugänglichen Geräten beides anbringt, kommt im Ernstfall an jedes Gerät heran. Ein NFC-Tag **ersetzt** den Aufkleber also nicht, er ergänzt ihn.

> **Wichtig:** Die Kennung eines NFC-Chips ist keine Sicherheitsangabe. Sie steht unverschlüsselt im Chip, jedes Handy liest sie, und beschreibbare Chips lassen sich mit beliebiger Kennung kaufen. Sie ist eine Kennung wie eine aufgedruckte Inventarnummer – nicht mehr. Was jemand nach dem Scannen sehen und ändern darf, entscheidet unverändert seine Berechtigung.

13.2 Aussehen des Etiketts festlegen

Einstellungen → Etiketten-Vorlage bestimmt, **was auf dem Etikett steht**: Barcode-Art (QR, DataMatrix oder Code 128), Layoutvorlage für den A4-Bogen, sichtbare Felder (Name, Seriennummer, CI-Klasse) und ob Ihr Logo aufgedruckt wird. Es gibt genau eine Vorlage je Mandant. Die Schaltfläche **Vorschau** erzeugt einen Beispielbogen aus den aktuellen – auch noch nicht gespeicherten – Einstellungen.

13.3 Etikettendrucker und Medien

> 💻 **Nur im Client:** *Fußbereich ▸ Einstellungen ▸ Etikettendrucker & Medien* · 🔑 `cmdb.class.manage` — Im Browser gibt es keinen Zugriff auf die lokale Druckerschnittstelle. Das Portal verwaltet unter *Verwaltung ▸ Etiketten* nur die Etiketten-Vorlagen, nicht die Geräte.

Die Einstellung beschreibt, **worauf gedruckt wird**. Davon dürfen Sie mehrere anlegen, denn ein Haus hat oft mehrere Geräte gleichzeitig im Einsatz: ein 57 × 32 mm großes gestanztes Etikett am Lagerdrucker, ein 12-mm-Endlosband am Handgerät, ein breites Rack-Schild.

🔒 **Pflichtreihenfolge** — die **Druckersprache** bestimmt, welche der übrigen Angaben überhaupt gelten. Ein PDF-Bogen kennt weder Schwärzung noch Stanzabstand.

1. Öffnen Sie *Einstellungen ▸ Etikettendrucker & Medien* und legen Sie ein Medium an.
2. Wählen Sie die **Druckersprache**.
3. Tragen Sie **Breite** und **Höhe** in Millimetern ein – Höhe `0` für Endlosmaterial.
4. Setzen Sie die **Auflösung** laut Datenblatt Ihres Druckers.
5. Feinen Sie bei Bedarf **Stanzabstand** und **Schwärzung** ab.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6. Setzen Sie *Als Vorgabe verwenden*, wenn dieses Medium ohne ausdrückliche Auswahl gelten soll.

✅ **Ergebnis:** Das Medium steht beim Drucken zur Auswahl. Je Medium legen Sie fest:

Angabe	Bedeutung
Druckersprache	PDF-Bogen (kein Etikettendrucker nötig), Zebra (ZPL II), Brother (P-touch/QL) oder Epson (ESC/POS)
Breite / Höhe in mm	Die Maße des Materials. **Höhe 0 bedeutet Endlosmaterial**
Auflösung	180, 203, 300 oder 600 dpi – steht im Datenblatt Ihres Druckers
Stanzabstand, Schwärzung	Feinabstimmung; Thermodirektpapier braucht weniger Schwärzung als Thermotransfer
Als Vorgabe verwenden	Dieses Medium wird ohne ausdrückliche Auswahl verwendet. Es gibt höchstens eines

Achtung: Tragen Sie Endlosband nicht als gestanztes Material ein (also Höhe 0 statt einer Zahl). Sonst sucht der Drucker nach einer Stanzung, die es nicht gibt, und zieht das Band durch.

Testdruck erzeugen legt einen Auftrag mit Musteretiketten an – ausdrücklich auch für ein Medium, das Sie noch nicht gespeichert haben. Die Maße stimmen selten beim ersten Anlauf.

Wichtig: Ordivis Platform **erzeugt** den Datenstrom für den Drucker, es druckt nicht selbst. Sie erhalten eine Datei und geben sie roh an das Gerät weiter (zum Beispiel durch Kopieren auf die Druckerfreigabe). Ordivis Platform baut von sich aus keine Verbindung zu Geräten in Ihrem Netz auf.

Was ein Gerät nicht kann, sagt Ordivis Ihnen. Nicht jede Druckersprache beherrscht alles; statt stillschweigend etwas anderes zu drucken, erscheint ein Hinweis am Auftrag:

- **Epson (ESC/POS)** kennt kein DataMatrix. Ist es in der Vorlage eingestellt, wird QR gedruckt.
- **Brother** druckt nur Punktmuster und kann keine Umlaut-Punkte darstellen; Umlaute werden umschrieben („Büro“ → „Buero“). Außerdem muss die eingetragene Breite der **Druckkopfbreite** Ihres Modells entsprechen, sonst druckt das Gerät versetzt.
- Ein **Logo** wird nur auf dem PDF-Bogen gedruckt. Bei Etikettendruckern müsste es vorab als Grafik in das Gerät geladen werden – das ist eine Einrichtung je Gerät.

Achtung: Die Ausgabe für Zebra, Brother und Epson folgt dem jeweils dokumentierten Befehlssatz und ist maschinell geprüft. Sie wurde jedoch **nicht an einem echten Gerät jedes Herstellers abgenommen**. Führen Sie vor dem Etikettieren größerer Bestände einen Testdruck durch.

13.4 Etikett für ein Gerät drucken

Am CI-Steckbrief erzeugt **Etikett drucken** den Auftrag für dieses Gerät. Haben Sie mehrere Medien angelegt, werden Sie gefragt, welches gemeint ist – wer zwei Drucker betreibt, meint mal den einen und mal den anderen. Ist kein Medium angelegt, erhalten Sie den A4-Etikettenbogen als PDF.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

13.5 NFC-Tag zuordnen

Am Steckbrief des Geräts in der Handy-App finden Sie den Abschnitt **NFC-Tags**. **Tag zuordnen und beschreiben** fordert Sie auf, den Tag an die Rückseite des Telefons zu halten; danach ist er dem Gerät zugeordnet. In einem Zug wird auf den Tag zusätzlich der geprüfte Verweis geschrieben – derselbe, der auch im QR-Code steht. Nur damit lässt sich später feststellen, dass der Tag aus Ihrem Ordivis Platform stammt.

Ein Gerät darf **mehrere** Tags tragen (Vorder- und Rückseite, Rack-Schiene). Hängt der Tag bereits an einem anderen Gerät, fragt die App nach, bevor sie ihn umhängt – beim Gerätetausch ist das richtig, beim versehentlichen zweiten Scannen im Lager wäre es der Verlust einer Zuordnung.

Hat Ihr Telefon keine NFC-Hardware, wird der Abschnitt gar nicht erst angezeigt.

13.6 Vor Ort scannen

Auf der Scan-Seite der App laufen **Kamera und NFC gleichzeitig**. Sie wählen keine Betriebsart, sondern halten hin, was da ist: Etikett vor die Kamera oder Tag an das Telefon. Beides landet in derselben Trefferliste und wird in der Inventur gleich verbucht.

Ist ein gelesener NFC-Tag noch keinem Gerät zugeordnet, ist das **kein Fehler** – beim ersten Kontakt mit einem frischen Tag ist es der Normalfall. Die App weist darauf hin, dass die Zuordnung am Steckbrief des Geräts erfolgt.

14 Datenqualität

Die **Datenqualitäts-Ansicht** zeigt Alterung (Aging) und Lücken (fehlende Pflichtattribute), damit die CMDB verlässlich bleibt.

Abbildung 6 zeigt diese Ansicht.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

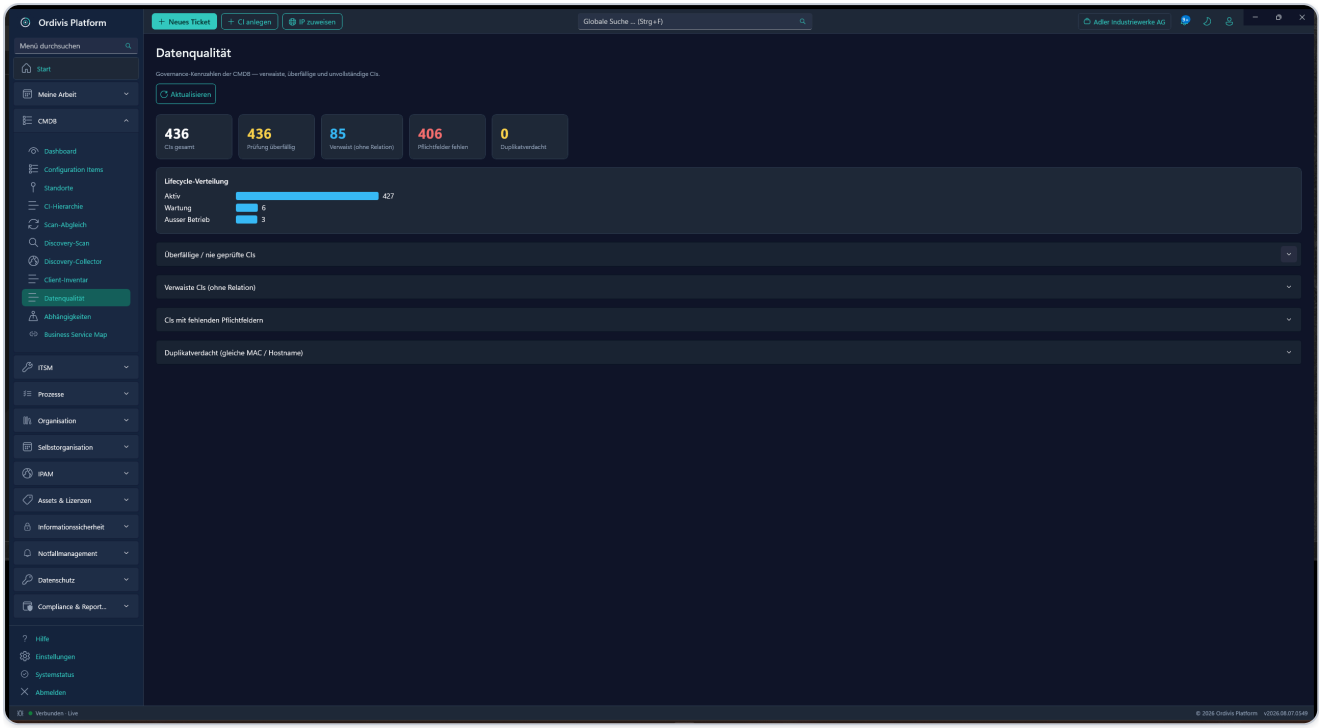

Abb. 6 – Datenqualität: überfällige Prüfungen, verwaiste CIs, fehlende Pflichtfelder und Duplikatverdacht.

15 Abgrenzung CMDB ↔ Asset Management

Ein Objekt kann zugleich **CI** (Betrieb, Relationen, Impact) und **Asset** (Kosten, Verträge, Garantie) sein. Beide Sichten werden **verknüpft**, nicht dupliziert. Details siehe ID-23.

16 Rollen & Rechte (Auszug)

Recht	Erlaubt
`cmdb.ci.read`	CIs ansehen
`cmdb.ci.create` / `cmdb.ci.update` / `cmdb.ci.delete`	CIs anlegen, bearbeiten, löschen
`cmdb.class.manage`	CI-Klassen und Attributkatalog verwalten
`cmdb.relation.manage`	Beziehungen zwischen CIs pflegen
`cmdb.import` / `cmdb.export`	Massenimport und Ausgabe

Verwandte Themen

- **ID-21a – Datenimport – Geschäftsprozesse** – Geschäftsprozesse per Datei übernehmen
- **ID-21b – Benutzerkonten in der CMDB** – Benutzerkonten als Configuration Item führen
- **ID-23 – Modul Asset- & Vertragsmanagement** – Abgrenzung zu Assets und Verträgen
- **ID-24 – Modul Discovery** – CIs automatisch aus dem Netz aufnehmen
- **ID-22 – Modul IPAM** – IP-Adressen und Subnetze am CI

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · PROZESSE

Datenimport – Geschäftsprozesse

Geschäftsprozesse per CSV/JSON importieren – Aufbau, Vorlagen, Beispiele

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-21a Import

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Datenimport – Geschäftsprozesse

Wenn Sie Ordivis Platform in Betrieb nehmen oder eine bestehende Prozesslandkarte übernehmen, möchten Sie selten jeden Prozess einzeln von Hand anlegen. Mit dem **Prozess-Import** übertragen Sie viele Geschäftsprozesse in einem Schritt aus einer Tabelle – als **CSV**- oder **JSON**-Datei. Eine solche Datei erstellen und pflegen Sie bequem in Excel, LibreOffice Calc oder jedem Texteditor.

Der Import befüllt den **Steckbrief** eines Prozesses (Name, Art, Ziel, Auslöser, Ergebnis, Verantwortungsrahmen, Verbesserungszyklus). Verknüpfungen zu anderen Objekten – die RACI-Matrix, die Kennzahlen, die Datenhoheit sowie die verantwortlichen Personen (Owner/Manager) – pflegen Sie weiterhin in der Anwendung selbst, weil sie auf andere Datensätze verweisen.

INHALT

1 Import starten
2 Aufbau der Datei
Zusatzfelder für die öffentliche Verwaltung
3 Neu anlegen oder aktualisieren (Upsert)
4 Vorschau und Fehler
5 Beispieldateien
CSV (Trennzeichen Semikolon, UTF-8)
JSON
6 Was der Import nicht abdeckt
Verwandte Themen

1 Import starten

💻 **Nur im Client:** *Prozesse* ▸ *Prozessübersicht* ▸ *Importieren* · 🔑 `cmdb.ci.update` — Das Portal zeigt die Prozesse unter *Prozesse* ▸ *Prozessübersicht* an, importiert aber nicht.

🔒 **Pflichtreihenfolge** — die Vorschau (Schritt 4) ist der Probelauf. Erst danach schreibt der Import. Wer Schritt 5 vorzieht, hat diese Kontrolle nicht.

1. Öffnen Sie in der Navigation den Bereich *Prozesse*.
2. Klicken Sie oben rechts auf *Importieren*.
3. Laden Sie über **CSV-Vorlage speichern…** oder **JSON-Vorlage speichern…** eine fertig vorbereitete Beispieldatei herunter und füllen Sie sie mit Ihren Prozessen.
4. Wählen Sie über **Datei wählen…** Ihre ausgefüllte Datei aus. Ordivis Platform zeigt sofort eine **Vorschau** (Probelauf): wie viele Prozesse neu angelegt bzw. aktualisiert würden und welche Zeilen fehlerhaft sind.
5. Stimmt die Vorschau, klicken Sie auf *Importieren*.

✅ **Ergebnis:** Die Prozesse stehen in der Prozessübersicht. Fehlerhafte Zeilen wurden **nicht** übernommen und stehen im Protokoll des Laufs.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Tipp:** In Excel speichern Sie die Datei über *Datei → Speichern unter* im Format **„CSV (Trennzeichen: Semikolon)“**. So bleiben Umlaute und Semikolons als Spaltentrenner erhalten.

2 Aufbau der Datei

Die **erste Zeile** einer CSV-Datei enthält die **Spaltennamen** (Kopfzeile), jede weitere Zeile einen Prozess. Bei JSON ist die Datei ein **Array von Objekten**, deren Feldnamen den Spaltennamen entsprechen. Groß-/Kleinschreibung der Spaltennamen spielt keine Rolle; nicht benötigte Spalten dürfen Sie weglassen.

Nur `process_name` ist ein Pflichtfeld. Alle übrigen Felder sind optional.

Feld (Spaltenname)	Pflicht	Bedeutung	Erlaubte Werte / Beispiel
`ci_number`	–	CI-Nummer eines **vorhandenen** Prozesses – nur zum Aktualisieren	z. B. `CI-000042`; zum Neuanlegen leer lassen
`process_name`	✓	Bezeichnung des Prozesses	z. B. `Rechnungsprüfung`
`process_type`	–	Prozessart	`Kernprozess`, `Unterstützungsprozess`, `Managementprozess`
`process_status`	–	Status (Freitext)	z. B. `Aktiv`, `Entwurf`
`fachaufgabe`	–	Fachaufgabe / Zweck	Freitext
`process_goal`	–	Ziel des Prozesses	Freitext
`process_trigger`	–	Auslöser	z. B. `Rechnung geht ein`
`process_result`	–	Ergebnis	z. B. `Zahlung angewiesen`
`process_input`	–	Input	Freitext
`process_output`	–	Output	Freitext
`detail_level`	–	Detaillierungsgrad	Freitext, z. B. `Grobprozess`
`capability`	–	Fähigkeit (Capability)	Freitext
`org_unit`	–	Organisationseinheit	z. B. `Finanzbuchhaltung`
`pdca_phase`	–	Phase im Verbesserungszyklus (PDCA)	`Plan`, `Do`, `Check`, `Act`
`review_interval_months`	–	Prüfintervall in Monaten	ganze Zahl, z. B. `12`
`last_review`	–	Datum der letzten Überprüfung	Format `JJJJ-MM-TT`, z. B. `2026-01-15`
`process_version`	–	Version des Steckbriefs	Freitext, z. B. `1.2`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld (Spaltenname)	Pflicht	Bedeutung	Erlaubte Werte / Beispiel
`valid_from`	–	Gültig ab	Format `JJJJ-MM-TT`
`valid_to`	–	Gültig bis	Format `JJJJ-MM-TT`
`legal_basis`	–	Rechtsgrundlage (Gesetze/Paragraphen)	Freitext, z. B. `§ 35 VwVfG`
`scope`	–	Geltungsbereich (räumlich/organisatorisch)	Freitext
`required_documents`	–	Erforderliche Unterlagen / Daten	Freitext
`recipient_type`	–	Leistungsempfänger	`Bürger`, `Unternehmen`, `Behörde`, `Intern`
`decision_rules`	–	Entscheidungskriterien / Ermessen	Freitext
`exceptions`	–	Ausnahmen & Fehlerpfade	Freitext
`media_breaks`	–	Medienbrüche	Freitext
`legal_deadline`	–	Gesetzliche Frist / Bearbeitungszeit	Freitext
`fees`	–	Gebühren & Kosten	Freitext

Wichtig: Datumsangaben (`last_review` , `valid_from` , `valid_to`) müssen im Format `JJJJ-MM-TT` vorliegen (z. B. `2026-01-15`). Andere Schreibweisen wie `15.01.2026` werden abgewiesen. Aus `last_review` und `review_interval_months` berechnet Ordivis Platform automatisch die nächste fällige Überprüfung.

Zusatzfelder für die öffentliche Verwaltung

Für Mandanten mit der **Organisationsart „Öffentliche Verwaltung"** (siehe Kapitel *CMDB → Prozessmanagement*) stehen zusätzliche Spalten bereit. Nutzen Sie sie nur, wenn Sie eine Verwaltungsleistung beschreiben; für andere Mandanten lassen Sie sie weg (der Import speichert sie zwar bei jedem Mandanten, angezeigt werden sie aber nur bei Verwaltungsmandanten).

Feld (Spaltenname)	Bedeutung	Erlaubte Werte / Beispiel
`leika_id`	LeiKa-ID / -Schlüssel	Ziffernschlüssel, z. B. `99123456789012`
`fim_id`	FIM-ID (Stamm-/Lokalprozess)	Freitext
`kgst_product`	KGSt-Produktnummer	Freitext
`ozg_service_field`	OZG-Themenfeld	eines der 14 OZG-Themenfelder, z. B. `Bauen & Wohnen`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld (Spaltenname)	Bedeutung	Erlaubte Werte / Beispiel
`ozg_id`	OZG-Leistungs-ID	Freitext
`egov_maturity`	Reifegrad (E-Government)	`Information`, `Kommunikation`, `Interaktion`, `Transaktion`
`trust_level`	Vertrauensniveau (eIDAS)	`Basis`, `Substantiell`, `Hoch`
`federal_level`	Föderalismus-Ebene	`Bund`, `Land`, `Kommunalverband`, `Gemeinde/Stadt`, `Mittelbehörde`
`legal_framework`	Rechtskreise & Ausführungsverordnungen	Freitext
`admin_act`	Verwaltungsakt (§ 35 VwVfG)	`Ja`, `Nein`

3 Neu anlegen oder aktualisieren (Upsert)

Ordivis Platform entscheidet je Zeile, ob ein Prozess **neu angelegt** oder ein **vorhandener aktualisiert** wird:

- Ist `ci_number` gefüllt, wird der Prozess mit **dieser CI-Nummer** aktualisiert. Existiert keiner mit dieser Nummer, meldet Ordivis Platform einen Fehler (Zeile wird übersprungen).
- Ist `ci_number` **leer**, sucht Ordivis Platform nach einem vorhandenen Prozess mit **exakt demselben Namen**. Wird genau einer gefunden, wird er aktualisiert; andernfalls entsteht ein **neuer** Prozess.

Beim Aktualisieren wirken nur die **in der Datei enthaltenen Spalten**: Ein gefüllter Wert überschreibt das Feld, eine **leere** Spalte **löscht** es, und eine **ganz fehlende** Spalte lässt das Feld unverändert. Neue Prozesse erhalten ihre CI-Nummer automatisch.

> **Achtung:** Der Name identifiziert einen Prozess nur, wenn er eindeutig ist. Tragen Sie zum gezielten Aktualisieren die `ci_number` ein – Sie finden sie in der Prozessliste bzw. am Steckbrief.

4 Vorschau und Fehler

Vor dem Schreiben führt Ordivis Platform immer einen **Probelauf** aus und zeigt, wie viele Prozesse **neu**, wie viele **aktualisiert** und wie viele **übersprungen** würden. Fehlerhafte Zeilen werden einzeln mit Zeilennummer, betroffenem Feld und Grund aufgeführt (z. B. „Zeile 4 · process_type: Ungültige Prozessart“). Korrigieren Sie die Datei und wählen Sie sie erneut aus. Der eigentliche Import schreibt nur die gültigen Zeilen; fehlerhafte werden übersprungen und stören die übrigen nicht.

5 Beispieldateien

Die folgenden Beispiele entsprechen exakt den Vorlagen, die Sie im Dialog herunterladen können. Die erste Zeile ist vollständig ausgefüllt, die zweite zeigt einen minimalen Prozess (nur Name und Art).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

CSV (Trennzeichen Semikolon, UTF-8)

```
ci_number;process_name;process_type;process_status;fachaufgabe;process_goal;process_trigger;
;Rechnungsprüfung;Kernprozess;Aktiv;Eingehende Rechnungen prüfen und freigeben;Fristgerechte
;Störungsannahme;Kernprozess;;;;;;;;;;;;
```

JSON

```
[
  {
    "process_name": "Rechnungsprüfung",
    "process_type": "Kernprozess",
    "process_status": "Aktiv",
    "fachaufgabe": "Eingehende Rechnungen prüfen und freigeben",
    "process_goal": "Fristgerechte, korrekte Zahlung",
    "process_trigger": "Rechnung geht ein",
    "process_result": "Zahlung angewiesen",
    "process_input": "Rechnung (PDF)",
    "process_output": "Zahlungsdatensatz",
    "detail_level": "Grobprozess",
    "capability": "Finanzwesen",
    "org_unit": "Finanzbuchhaltung",
    "pdca_phase": "Do",
    "review_interval_months": "12",
    "last_review": "2026-01-15"
  },
  {
    "process_name": "Störungsannahme",
    "process_type": "Kernprozess"
  }
]
```

Tipp: Um bestehende Prozesse zu aktualisieren, ergänzen Sie in einer Zeile die Spalte `ci_number` mit der CI-Nummer des Prozesses und passen die gewünschten Felder an.

6 Was der Import nicht abdeckt

Der Import befüllt ausschließlich den **Steckbrief**. Die folgenden Angaben pflegen Sie weiterhin direkt am Prozess in der Anwendung, weil sie auf andere Datensätze verweisen:

- **Prozessverantwortliche/r** und **Prozessmanager/in** (Verweis auf eine Person),
- die **RACI-Matrix** (Rollen/Benutzer),
- die **Kennzahlen** (KPIs),
- die **Datenhoheit** (CRUD-Matrix) und die **Verknüpfungen** zu Systemen, Schnittstellen, Lieferanten und Kunden,
- das grafische **Prozessmodell (BPMN)**, das Sie auf der Registerkarte *Prozessmodell* zeichnen oder als BPMN-/XML-Datei importieren.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Verwandte Themen

- **ID-21 – Modul CMDB** – wohin die importierten Objekte gehören
- **ID-33 – Modul Organisation & Geschäftsverteilungsplan** – Prozesse, Federführung und Geschäftsverteilungsplan
- **ID-29 – Modul Business Continuity Management** – Prozesse als Grundlage der Business-Impact-Analyse

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · CMDB

Benutzerkonten in der CMDB

Benutzer aus Active Directory und lokale Konten automatisch als Configuration Item führen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-21b Benutzerkonten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Benutzerkonten in der CMDB

Ordivis Platform führt jedes **Benutzerkonto** – aus dem **Active Directory (AD/LDAP)** wie auch **lokal** in Ordivis Platform angelegt – automatisch als Configuration Item der Klasse **„Benutzerkonto"**. So erscheinen Ihre Anwenderinnen und Anwender ohne manuelle Pflege in der CMDB und lassen sich mit Geräten, Diensten, Rollen und Prozessen verknüpfen (z. B. „Konto *ist zugewiesen* Notebook", „Konto *ist Mitglied von* Gruppe").

Die Übernahme ist eine **reine Spiegelung**: Ordivis Platform schreibt niemals ins Verzeichnis zurück, sondern bildet den aktuellen Stand der Benutzertabelle im CI ab.

INHALT

1 Wie die Übernahme funktioniert
2 Welche Felder gespiegelt werden
3 Abgrenzung zu ähnlichen Klassen
4 Voraussetzungen und Betrieb
4.1 Ein Benutzerkonto in der CMDB nachschlagen
4.2 Die Felder der Klasse ansehen
Verwandte Themen

1 Wie die Übernahme funktioniert

Ein Hintergrunddienst gleicht in regelmäßigen Abständen (Standard: alle **30 Minuten**) alle Benutzer eines Mandanten mit der CMDB ab:

- **Neues Konto** → es wird ein neues „Benutzerkonto"-CI angelegt (mit automatischer CI-Nummer).
- **Geändertes Konto** → das vorhandene CI wird aktualisiert.
- **Deaktiviertes oder entferntes Konto** → das CI wird auf den Lebenszyklus-Status **„Außer Betrieb"** gesetzt (es bleibt für Nachweise/Verknüpfungen erhalten, wird aber nicht gelöscht).

Wiedererkannt wird ein Konto über eine stabile interne Kennung; ein Konto führt also **genau ein** CI – auch nach Namens- oder Abteilungswechsel. **Systemkonten** von Ordivis Platform werden bewusst **nicht** übernommen.

> **Hinweis:** Weil es sich um eine Spiegelung handelt, werden **manuelle Änderungen** an den gespiegelten Feldern eines Benutzerkonto-CIs beim nächsten Abgleich wieder **überschrieben**. Pflegen Sie solche Angaben an der Quelle – im AD bzw. in der Benutzerverwaltung.

2 Welche Felder gespiegelt werden

Bei **lokalen** Konten werden die in Ordivis Platform gepflegten Stammdaten übernommen. Bei **AD-/LDAP**-Konten liest Ordivis Platform beim Verzeichnis-Abgleich **alle** lesbaren AD-Attribute aus und trägt sie in die benannten Felder ein; der vollständige Rohsatz steht zusätzlich im Feld **„Alle AD-Attribute (roh)"**.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Abschnitt	Felder (Auswahl)	Quelle im AD
Konto	Herkunft (AD/LDAP oder Lokal), Anmeldename, UPN, Aktiviert, AD-Objekt-GUID, Distinguished Name	`sAMAccountName`, `userPrincipalName`, `userAccountControl`, `objectGUID`, `distinguishedName`
Name	Anzeigename, Vorname, Nachname, Initialen	`displayName`, `givenName`, `sn`, `initials`
Organisation	Abteilung, Position/Titel, Firma, Büro, Personalnummer, Mitarbeitertyp, Vorgesetzte/r	`department`, `title`, `company`, `physicalDeliveryOfficeName`, `employeeID`, `employeeType`, `manager`
Kontakt	E-Mail, Telefon, Mobil, Fax, Webseite	`mail`, `telephoneNumber`, `mobile`, `facsimileTelephoneNumber`, `wWWHomePage`
Adresse	Straße, PLZ, Ort, Bundesland, Land	`streetAddress`, `postalCode`, `l`, `st`, `co`
Verzeichnis	Gruppen (memberOf), AD angelegt/geändert am, letzte Anmeldung, Konto läuft ab, **Alle AD-Attribute (roh)**	`memberOf`, `whenCreated`, `whenChanged`, `lastLogonTimestamp`, `accountExpires`

Bei lokalen Konten bleiben die reinen AD-Felder (Adresse, Büro, Gruppen, Rohsatz usw.) leer – gefüllt werden Anmeldename, Anzeigename, Vor-/Nachname, E-Mail, Abteilung und Telefon, soweit vorhanden.

3 Abgrenzung zu ähnlichen Klassen

Ordivis Platform trennt bewusst drei Sichten auf „Personen und Konten":

- **Benutzerkonto** (dieses Kapitel) – der **automatische Spiegel** eines Anmeldekontos aus AD oder Ordivis Platform. Wird maschinell gepflegt.
- **Person (intern)** – **manuell** gepflegte Personal-/Stammdaten einer Mitarbeiterin oder eines Mitarbeiters (z. B. Eintrittsdatum, zugeordneter Arbeitsplatz). Wird vom Konto-Sync **nicht** angetastet.
- **Digitale Identität** – die **Governance-Sicht** auf besondere Konten (Dienst-, Technik-, Notfall- konten) mit Rezertifizierung und Kennzeichnung „privilegiert".

4 Voraussetzungen und Betrieb

- Für die Übernahme von **AD-Benutzern** muss die **LDAP-Anbindung** eingerichtet und aktiv sein (siehe Kapitel *Identität, Benutzer & Rollen*, ID-41). Der Verzeichnis-Abgleich liest die Konten samt Attributen; der anschließende CMDB-Abgleich überträgt sie in Benutzerkonto-CIs.
- **Lokale Konten** benötigen keine weitere Einrichtung – sie werden automatisch mitgeführt.

4.1 Ein Benutzerkonto in der CMDB nachschlagen

💻 **Client:** *CMDB* ▸ *Configuration Items* · 🌐 **Portal:** *CMDB* ▸ *CMDB-Suche* · 🔑 `cmdb.ci.read`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Öffnen Sie die CI-Liste.
2. Filtern Sie auf die Klasse **Benutzerkonto**.
3. Suchen Sie das Konto über Anmeldename oder Anzeigenamen.
4. Öffnen Sie das CI, um Konto-, Namens-, Organisations-, Kontakt- und Verzeichnisdaten zu sehen.

✅ **Ergebnis:** Sie sehen den gespiegelten Stand des Kontos samt seiner Verknüpfungen zu Geräten, Diensten und Gruppen.

4.2 Die Felder der Klasse ansehen

> 💻 **Client:** *Fußbereich* ▸ *Einstellungen* ▸ *CI-Klassen* · 🌐 **Portal:** *Verwaltung* ▸ *CI-Klassen* · 🔑 `cmdb.class.manage`

Die Klasse „Benutzerkonto" und ihre Felder finden Sie wie jede andere CI-Klasse dort. Die angelegten Konten erscheinen in der CI-Liste und lassen sich filtern, exportieren und verknüpfen.

> **Achtung:** Die gespiegelten Felder sind **nicht** zum Bearbeiten gedacht. Änderungen daran werden beim nächsten Abgleich überschrieben – siehe den Hinweis in Abschnitt 1. Wer die Angabe ändern will, ändert sie im AD bzw. in der Benutzerverwaltung.

Verwandte Themen

- **ID-21 – Modul CMDB** – die CMDB, in der die Konten geführt werden
- **ID-41 – Modul Identität, Benutzer & Rollen** – Anmeldung, Rollen und Rechte derselben Benutzer
- **ID-24 – Modul Discovery** – woher die Konten aus dem Verzeichnisdienst kommen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · CMDB

Rechenzentrums-Dokumentation

Gerätemodelle, Komponenten, Patchfelder, Wegverfolgung und Rackbelegung

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-21c Rechenzentrum

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Rechenzentrums-Dokumentation

Die CMDB beantwortet, **was** Sie betreiben und **wovon es abhängt**. Dieses Kapitel beantwortet die Fragen davor: **Was steht wo im Schrank, was steckt daran, und wo führt das Kabel hin?**

Das klingt nach Kleinkram, bis der erste Handgriff ansteht. Ein Techniker steht vor zwölf Schränken und sucht den Server, der ausgefallen ist. Ein Kollege plant einen neuen Datenbankserver und sucht vier freie Höheneinheiten. Ein dritter verfolgt eine Leitung von einem Switch-Port über zwei Patchfelder bis in ein Büro im dritten Stock. Ohne erfasste physische Schicht beginnt jede dieser Aufgaben mit einer Runde durch den Serverraum — und endet gelegentlich damit, dass jemand am falschen Gerät zieht.

Ordivis Platform bildet dafür vier Dinge ab:

Was	Wozu
Gerätemodelle	„Was ist das für ein Gerät?" — Bauhöhe, Bautiefe, Luftführung, Gewicht, Anschlussbestückung
Komponenten	„Was steckt daran?" — Netzwerkschnittstellen, Konsolen-, Strom- und Patchfeld-Anschlüsse
Rackbelegung	„Wo sitzt es?" — Höheneinheit, Seite, Bautiefe, freie und vorgemerkte Plätze
Wegverfolgung	„Wo führt das hin?" — die Spur von Anschluss zu Anschluss über beliebig viele Patchfelder

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

1 Der Gerätemodell-Katalog
- 1.1 Warum ein Modell kein Configuration Item ist
- 1.2 Was ein Modell festhält
- 1.3 Der ausgelieferte Grundbestand
- 1.4 Herkunft — und warum sie an jedem Eintrag steht

2 Zusätzliche Gerätemodelle einlesen

3 Bestückungsvorlagen: der eigentliche Hebel
- 3.1 Ein Modell einem Gerät zuordnen
- 3.2 Anschlüsse als Reihe anlegen

4 Komponenten am Gerät
- 4.1 Verbindungen

5 Patchfelder und Durchgänge

6 Wegverfolgung

7 Rackbelegung, Bautiefe und Vormerkungen
- 7.0 Der Belegungsplan im Client
- 7.1 Bautiefe
- 7.2 Höhenbereiche vormerken
- 7.3 Rackmodelle
- 7.4 Massenpflege der Einbaupositionen
- 7.5 Der Rack-Marker für die überlagerte Ansicht

8 Übernahme aus einer NetBox-Instanz
- 8.1 Es ist eine Migration, keine Dauersynchronisation
- 8.2 Zuordnung der CI-Klassen
- 8.3 Vorgehen
- 8.4 Die Falle bei der Einbauposition

9 Rollen & Rechte (Auszug)

Verwandte Themen

1 Der Gerätemodell-Katalog

Ein **Gerätemodell** beschreibt einen Gerätetyp, nicht ein einzelnes Gerät: „Dell PowerEdge R740" ist ein Modell, der Server mit der Seriennummer *ABC123* ist ein Configuration Item, das auf dieses Modell verweist.

Sie finden den Katalog im Menü unter **CMDB → Gerätemodell-Katalog**.

1.1 Warum ein Modell kein Configuration Item ist

Naheliegend wäre, Gerätemodelle einfach als weitere CI-Klasse zu führen. Das wäre falsch, und zwar aus zwei Gründen, die beide erst später wehtun:

- Ein Modell hat **keinen Lebenszyklus, keine Zuweisung und keinen Standort**. Als CI bekäme es all das aufgezwungen — auch die Pflicht, in einem Raum zu stehen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- Jede **Bestandszählung** zählte die Katalogeinträge mit. Auf die Frage „wie viele Geräte haben wir?" käme dann eine Zahl heraus, die 4.000 Modellbeschreibungen enthält.

Der Katalog ist deshalb ein **Verzeichnis neben** dem Bestand, so wie Etikettenvorlagen und Nummernkreise. Ein CI verweist optional darauf.

1.2 Was ein Modell festhält

Angabe	Bedeutung
Hersteller / Modell	Die Bezeichnung, unter der Sie das Gerät bestellen
Bestellnummer	Die SKU des Herstellers, falls sie abweicht
Bauhöhe	In Höheneinheiten, **halbe Einheiten inbegriffen** (0,5 HE gibt es wirklich)
Bautiefe	Volltief oder halbtief — ein volltiefes Gerät belegt beide Seiten des Höhenrasters
Luftführung	Vorn → hinten, hinten → vorn, seitlich, passiv … — die Angabe für die Kaltgangplanung
Gewicht	Für die Traglastrechnung je Schrank
Leistungsaufnahme	Für die Strombedarfsrechnung je Schrank
Bestückung	Die Anschlüsse, die ein Gerät dieses Modells mitbringt (siehe Abschnitt 3)

> **Einschübe belegen keine Höhe.** Eine Blade oder eine Einschubkarte sitzt im Gehäuse, nicht im Rack. Solche Modelle tragen die Bauhöhe 0 und tauchen im Belegungsplan nicht auf — sonst rechnete er den Platz doppelt.

1.3 Der ausgelieferte Grundbestand

Der Katalog ist **beim ersten Start bereits gefüllt**: über 4.000 Gerätemodelle von 35 Herstellern, dazu rund 140 Rackmodelle — jeweils samt Anschlussbestückung.

Das ist bewusst so gelöst und kein Zufall. Ordivis Platform wird überwiegend in Kommunen und Behörden betrieben, und deren Serverumgebungen haben häufig **keinen ausgehenden Internetzugang**. Ein Katalog, der erst geladen werden müsste, bliebe dort dauerhaft leer — und mit ihm die halbe Funktion dieses Kapitels. Der Grundbestand wird deshalb mit dem Produkt ausgeliefert und über das gewohnte Produktupdate aktualisiert.

Die Auswahl folgt drei Regeln:

1. **Hersteller nach Verbreitung im Zielmarkt** — Server, Netzwerk, Sicherheit, Strom und unterbrechungsfreie Versorgung, Speicher, Konsolenzugriff, Verteiler und Schränke.
2. **Innerhalb eines gewählten Herstellers keine weitere Auswahl.** Sonst müsste jemand entscheiden, welcher Switch „wichtig genug" ist — und genau das eine Modell fehlte dann bei Ihnen.
3. **Ohne Geräteansichten.** Fotos und Frontansichten machen den weit überwiegenden Teil der Quelle aus und tragen zur Bestandsführung nichts bei; mit ihnen wäre der Installer unbrauchbar groß.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1.4 Herkunft — und warum sie an jedem Eintrag steht

Jeder Katalogeintrag trägt sichtbar seine Herkunft:

Herkunft	Bedeutung
ausgeliefert	Kam mit dem Produkt. Nicht änderbar.
eingelesen	Aus einer Modelldatei übernommen (Abschnitt 2).
selbst angelegt	Von Hand erfasst.

Die Regel dahinter ist einfach und wichtig: **Ein Produktupdate fasst ausschließlich ausgelieferte Einträge an.** Was Sie selbst eingelesen oder angelegt haben, bleibt unangetastet — auch dann, wenn es denselben Kurznamen trägt.

> **Warum ausgelieferte Einträge nicht änderbar sind.** Eine Änderung daran wäre beim nächsten Update wieder weg, und das sähe wie ein Fehler des Produkts aus. Wenn ein ausgeliefertes Modell nicht zu Ihrem Gerät passt, legen Sie ein eigenes an — das gehört dann Ihnen und bleibt.

---

2 Zusätzliche Gerätemodelle einlesen

Kein noch so guter Grundbestand deckt jedes Gerät ab. Über **Modelldateien einlesen …** übernehmen Sie weitere Modelle aus Definitionsdateien im YAML-Format — eine einzelne Datei oder einen ganzen Ordner.

Der Ablauf ist bewusst zweistufig:

1. Sie wählen die Dateien aus. Ordivis Platform führt **zuerst einen Probelauf** durch und meldet, wie viele Modelle neu wären, wie viele aktualisiert und wie viele übersprungen würden.
2. Erst danach fragt der Dialog, ob wirklich geschrieben werden soll.

Wer dreihundert Dateien auf einmal einliest, beantwortet die Frage „berührt das meinen Bestand?" vorher leichter als hinterher. Nach dem Lauf bleibt die **Restliste** stehen: jede Datei mit Ergebnis und Grund. So hat die Nacharbeit eine Liste und keine Suche.

Was der Leser nicht versteht, wird **gemeldet und nicht überlesen**. Das ist Absicht: Ein Import, der unbekannte Zeilen stillschweigend verwirft, erzeugt ein Modell mit halber Bestückung — und das sieht aus wie ein gepflegtes.

---

3 Bestückungsvorlagen: der eigentliche Hebel

Ein Katalog, der nur Bauhöhen kennt, spart Ihnen ein paar Minuten. Der Nutzen entsteht erst mit der **Bestückung**: Das Modell führt seine Anschlüsse mit, und ein Gerät erbt sie beim Anlegen.

48 Ports tippt niemand von Hand. Ohne Vorlagen bleibt jede Portverwaltung eine Absichtserklärung, die einmal befüllt und danach nie wieder angefasst wird — und eine Verkabelungsdokumentation, der man

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

nicht glaubt, ist schlimmer als keine, weil trotzdem jemand nach ihr greift.

3.1 Ein Modell einem Gerät zuordnen

Sie öffnen ein Gerät in der CMDB und wählen sein Modell. Ordivis Platform legt daraufhin die Anschlüsse an, die das Modell mitbringt.

- Ein **Probelauf** zeigt vorher, was entstünde und was unangetastet bliebe.
- **Vorhandene Anschlüsse werden nie überschrieben.** An ihnen hängen möglicherweise bereits gesteckte Verbindungen und zugewiesene IP-Adressen. Sie bleiben stehen und werden im Bericht genannt.
- Hersteller und Modell des Geräts kommen ab dann **aus dem Katalog**. Wer die Felder frei setzen will, löst die Modellzuordnung.

3.2 Anschlüsse als Reihe anlegen

Auch ohne Modell müssen Sie Ports nicht einzeln tippen. Bei der Eingabe geben Sie ein Namensmuster mit dem Platzhalter `{n}` und einen Bereich an:

Muster	Bereich	Ergebnis
`GigabitEthernet1/0/{n}`	`1–48`	48 Anschlüsse, fortlaufend benannt
`Front {n}`	`1–24`	24 Front-Ports eines Patchfelds

Bereits vergebene Namen werden übersprungen, nicht abgewiesen: Wer eine Reihe ein zweites Mal anlegt, weil ihm vier Ports fehlten, bekommt die vier.

---

4 Komponenten am Gerät

Bis hierher kannte die CMDB nur **Netzwerkschnittstellen**. Ein Gerät hat aber mehr Anschlüsse als die, und die anderen sind für den Handgriff vor dem Schrank oft wichtiger. Ordivis Platform führt deshalb **Geräte-Komponenten** in diesen Arten:

Art	Was es ist
Netzwerk	Port oder Netzwerkkarte
Konsole	Serieller oder USB-Anschluss am verwalteten Gerät
Konsolen-Server	Die bedienende Seite — Terminalserver, KVM
Strom-Eingang	Was das Gerät aufnimmt (Kaltgerätestecker am Netzteil)
Strom-Abgang	Was das Gerät abgibt (Buchse an Steckdosenleiste oder USV)
Front-Port	Vorderseite eines Patchfeld-Durchgangs
Rear-Port	Rückseite eines Patchfeld-Durchgangs
Modulschacht	Steckplatz für eine Einschubkarte

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Jede Komponente trägt ihren **Typ** aus einer geprüften Werteliste: Schnittstellenarten (`10gbase-t`, `1000base-x-sfp`, …), Steckverbinder (`lc-apc`, `8p8c`, `mpo`, …), Stromanschlüsse (`iec-60320-c13`, `nema-l6-30r`, …) — insgesamt über 400 Werte.

> **Warum geprüft und nicht frei getippt.** Ein freies Textfeld für den Steckertyp führt binnen eines Jahres zu „LC", „lc", „LC-Duplex" und „LC/PC" nebeneinander. Eine Auswertung zählt dann vier Sorten Glasfaser, wo eine steht.

4.1 Verbindungen

Zwei Anschlüsse werden miteinander **verbunden** — reziprok, also von beiden Seiten sichtbar. Eine Verbindung trägt einen Zustand:

Zustand	Bedeutung
gesteckt	Das Kabel liegt.
geplant	Die Zuordnung steht, das Kabel liegt noch nicht.
im Rückbau	Noch gesteckt, zur Auflösung vorgesehen.

Zusätzlich lässt sich ein Anschluss als **belegt, Gegenstelle nicht erfasst** kennzeichnen — etwa der Anschluss ans Hausnetz oder ein Bauteil im Fremdbereich. Ohne diese Angabe stünde derselbe Port in jeder Auswertung als frei und würde ein zweites Mal vergeben.

---

5 Patchfelder und Durchgänge

Ein **Patchfeld** ist kein Gerät mit Schnittstellen, sondern ein **Durchgang**: vorne gesteckt, hinten fest verlegt. Die eigentliche Information ist die Zuordnung *Front-Port → Rear-Port*. Ordivis Platform führt Patchfelder deshalb als eigene CI-Klasse.

Ein Rear-Port kann mehrere **Positionen** tragen. Bei einer MPO-Buchse mit zwölf Fasern sind das zwölf: Eine Rückseite führt dort tatsächlich zu zwölf vorderen Anschlüssen.

---

6 Wegverfolgung

Zu jedem Anschluss lässt sich der **Weg zur Gegenstelle** verfolgen — über beliebig viele Patchfelder hinweg. Das Ergebnis nennt jede Station mit Gerät und Anschlussnamen.

Die Spur endet auf eine von sechs Weisen, und jede sagt etwas anderes:

Ergebnis	Bedeutung
Weg vollständig	Die Spur endet an einem echten Anschluss der Gegenseite.
Nichts angeschlossen	Am Ausgangspunkt steckt nichts.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ergebnis	Bedeutung
Sackgasse	Die Spur bricht unterwegs ab — ein Durchgang führt ins Nichts.
Ringverkabelung	Der Weg läuft im Kreis.
Weg teilt sich	Eine mehrfach belegte Rückseite, ohne dass aus dem Verlauf hervorgeht, welche Faser gemeint ist.
Belegt, Gegenstelle nicht erfasst	Sie haben ausdrücklich erklärt, dass es hier nicht weitergeht.

Zwei Punkte, an denen die Anzeige bewusst zurückhaltend ist:

- **Eine geplante Teilstrecke macht den ganzen Weg zum Plan.** Wer „geplant" wie „gesteckt" behandelt, schickt jemanden an einen toten Anschluss.
- **Bei einer Aufteilung wird nicht geraten.** Eine geratene Faser führt den Techniker an den falschen Anschluss — und sie sieht dabei überzeugend aus.

---

7 Rackbelegung, Bautiefe und Vormerkungen

Der Belegungsplan eines Racks zeigt Vorder- und Rückseite mit ihrem Höhenraster, alle Einbauten mit ihrem HE-Bereich und die freien Plätze.

7.0 Der Belegungsplan im Client

> 💻 **Nur im Client:** *CMDB* ▸ *Configuration Items* ▸ *Rack-CI öffnen* ▸ *Rack-Belegungsplan* · 🔑 `cmdb.ci.read` zum Ansehen, `cmdb.ci.update` zum Ein- und Ausbauen — Das Portal zeigt Rack-CIs wie jedes andere CI, aber ohne Belegungsplan.

Die Schaltfläche *Rack-Belegungsplan* erscheint nur an Racks.

Die Ansicht zeigt genau **eine Seite** — Vorder- oder Rückseite, umschaltbar. Beide zusammen in einem Raster zu zeichnen wäre irreführend: Der Server führt sie als eigenständige Flächen, und ein Gerät auf HE 20 vorn hat mit HE 20 hinten nichts zu tun (außer bei volltiefen Geräten, siehe 7.1).

Über dem Raster steht die Bilanz der gezeigten Seite und die **größte zusammenhängende Lücke**. Die Gesamtzahl freier Einheiten allein hilft nicht weiter: Acht freie Einheiten, verteilt auf acht Einzellücken, nehmen keinen 4-HE-Server auf.

An jedem Einbau stehen zwei Schaltflächen:

Aktion	Wirkung
Verschieben	Unterste Höheneinheit und Höhe ändern. Der Kollisionshinweis erscheint **während** der Eingabe, nicht erst beim Speichern.
Ausbauen	Der Platz wird frei, das CI bleibt bestehen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Warum kein Ziehen und Ablegen. Eine Höheneinheit ist auf dem Schirm wenige Pixel hoch. Ein um eine Einheit verrutschter Einbau ist genau der Fehler, den dieser Plan verhindern soll — deshalb steht die Zahl im Dialog und lässt sich vor dem Speichern lesen.

⚠️ **Ist am Rack keine Höhe (`height_u`) gepflegt**, zeichnet die Ansicht kein Raster und sagt das. Eine geratene Höhe wäre eine Behauptung über einen Schrank, den niemand vermessen hat.

Geräte, die Sie nicht sehen dürfen (Berechtigung), belegen ihren Platz sichtbar, geben aber ihren Namen nicht preis. Sie ganz wegzulassen wäre schlechter: An einem lückenhaften Raster zählt ein Techniker falsch und greift daneben.

Aus `rack_unit` übernehmen wandelt das alte Freitextfeld der Geräte dieses Racks in echte Einbauten. Der Lauf zeigt erst eine Vorschau — er kann für ein volles Rack über vierzig Einbauten anlegen.

7.1 Bautiefe

Ein **volltiefes** Gerät belegt beide Seiten des Rasters. Ohne diese Angabe liesse die Kollisionsprüfung hinter einem volltiefen Server ein zweites Gerät auf derselben Höhe zu — physisch unmöglich, in der Belegungstabelle aber unauffällig.

Einbauten, die vor der Einführung dieser Angabe erfasst wurden, gelten als **nicht volltief**. Sie nachträglich als volltief zu behaupten liesse gepflegte Rückseiten-Einbauten plötzlich als Kollision erscheinen — an Schränken, an denen niemand etwas geändert hat.

7.2 Höhenbereiche vormerken

Ein **vorgemerkter** Höhenbereich ist Platz, der bereits vergeben ist, obwohl noch nichts darin steht. Das ist der praktische Kern dieses Abschnitts, denn ohne ihn passiert Folgendes, und zwar zuverlässig: Zwei Personen sehen in dieselbe Belegungstabelle, beide finden HE 20–23 frei, beide planen dorthin — und der Konflikt fällt am Liefertag auf, wenn beide Geräte im Haus sind.

Jede Vormerkung braucht einen **Zweck**. Eine Vormerkung ohne Zweck kann später niemand beurteilen und deshalb auch niemand auflösen.

Zustand	Wirkung
angemeldet	Blockiert bereits — sonst vergibt der Nächste den Platz, während die Bestätigung läuft.
gültig	Blockiert.
überfällig	Blockiert **nicht** mehr, bleibt aber sichtbar.

Belegt und **vorgemerkt** werden getrennt gemeldet und getrennt gezählt. Beides verlangt eine andere Reaktion: Beim Einbau weicht man aus, bei der Vormerkung ruft man den an, der sie eingetragen hat. Und für die Beschaffung ist der Unterschied zwischen „hier steht etwas" und „hier soll etwas hin" genau der Punkt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

7.3 Rackmodelle

Wie bei Gerätemodellen gibt es einen Katalog für **Rackmodelle** — Bauform, Nutzbreite, Höhe, unterste Höheneinheit, Zählrichtung, Außenmaße, Einbautiefe und Traglast. Ein Rackmodell ist eine **Vorlage**: Es füllt die Maße eines Schranks einmal aus, danach gehören sie dem Schrank. Ein nachträglich umgebauter Schrank ändert sie dort, ohne dass ein Katalogeintrag für vierzig andere mitwandert.

> **Zählrichtung.** Fast alle Schränke sind von unten nach oben beschriftet. Es gibt Ausnahmen, und die sind nicht verhandelbar: Zählt die Anzeige andersherum als der Aufkleber, sitzt jede Angabe sauber gespiegelt — und genau deshalb fällt es niemandem auf, bis jemand den falschen Server zieht.

7.4 Massenpflege der Einbaupositionen

Im Belegungsplan geben **Tabelle ausgeben** und **Tabelle einlesen** die erfassten Einbauten als CSV-Datei aus und nehmen sie wieder entgegen. Die Ausgabe umfasst **alle** Schränke, nicht nur den gerade geöffneten: Vierzig Racks einzeln zu exportieren wäre dieselbe Handarbeit, die die Funktion abnehmen soll. Die Spalten der Ausgabe sind genau die, die der Import liest:

```
rack_nummer;rack_name;geraet_nummer;geraet_name;start_he;hoehe_he;seite;volltief;ausrichtung
```

Zugeordnet wird über die **CI-Nummer**, nicht über den Namen — zwei Server heißen erstaunlich oft gleich. Die Klarnamen stehen nur zur Kontrolle daneben und werden beim Einlesen nicht ausgewertet. `seite` versteht *vorn* und *hinten* (englisch *front*/*rear*), `volltief` *ja* und *nein*; leer bedeutet Vorderseite bzw. nicht volltief.

Das Einlesen zeigt **zuerst eine Vorschau**. Sie rechnet vollständig — einschließlich der Kollisionen der Datei mit sich selbst — schreibt aber nichts. Erst die Bestätigung übernimmt.

> **Was der Import nicht tut: raten.** Eine unbekannte oder mehrfach vergebene CI-Nummer, eine unleserliche Höhenangabe, ein belegter Platz — jede dieser Zeilen steht mit Grund im Bericht, statt mit einer Annahme im Rack zu landen. Eine scheinbar gepflegte Belegung, in der Geräte an erfundenen Stellen sitzen, ist schlechter als die leere Tabelle, mit der man begonnen hat.
>
> **Vorhandene Einbauten werden nicht verschoben.** Steht ein Gerät bereits an anderer Stelle, meldet der Bericht das und lässt es stehen. Verschoben wird im Belegungsplan, wo man sieht, was man tut. Zeilen, die den bereits erfassten Zustand beschreiben, zählen als *unverändert* und erscheinen nicht im Bericht — sonst läse man nach dem Zurückspielen vierzig Meldungen über das, was ohnehin stimmt.

7.5 Der Rack-Marker für die überlagerte Ansicht

Die überlagerte Rack-Ansicht auf dem Mobilgerät legt die Belegung über das Kamerabild. Damit sie weiß, **wo** die Rackfront im Bild liegt und **wie groß** sie ist, braucht sie einen Bezugspunkt am Schrank: den **Rack-Marker**.

Der Marker ist **kein eigenes Format**. Es ist das reguläre CI-Etikett des Rack-Schranks — derselbe QR-Code mit Prüfwert, den auch jedes Gerät trägt (siehe *Etiketten und Kennzeichnung*). Gedruckt wird er über das

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

vorhandene Etikettensystem; ein gefälschter Aufkleber fällt damit genauso auf wie an einem Gerät.

Montage:

1. Das CI-Etikett des **Racks** drucken (nicht das eines eingebauten Geräts).
2. Den Aufkleber **auf die Frontblende** kleben — flach, nicht auf eine Tür mit Glaseinsatz und nicht über eine Kante. Ein geknickter oder gewölbter Marker liefert verzerrte Eckpunkte, und die Verzerrung wandert in die ganze Projektion.
3. Am Rack-CI unter **AR-Marker** drei Maße eintragen:

Feld	Bedeutung
`Marker: Kantenlänge (mm)`	Kantenlänge des **aufgedruckten QR-Quadrats**, nicht des Aufklebers
`Marker: Abstand von links (mm)`	von der linken Kante der Rackfront zur linken Markerkante
`Marker: Höhe über HE 1 (mm)`	von der **Unterkante der Höheneinheit 1** zur unteren Markerkante

> **Warum ab HE 1 und nicht ab Boden.** Schränke stehen auf unterschiedlich hohen Sockeln. Ein Maß ab Boden wäre bei jedem Modell ein anderes und ließe sich am Schrank nicht nachmessen; die Unterkante der ersten Höheneinheit ist dagegen abzulesen.
>
> **Nachmessen statt Umkleben.** Die drei Zahlen beschreiben den Zustand am Schrank, sie sind keine Montagevorgabe. Sitzt der Aufkleber drei Zentimeter neben der geplanten Stelle, wird hier die tatsächliche Lage eingetragen — die Projektion stimmt danach wieder. Ohne diese Werte am CI bliebe nur das Umkleben, an vierzig Schränken.

Solange eines der drei Maße oder die Rackhöhe fehlt, zeichnet die Ansicht **keine** Überlagerung. Das ist Absicht: Ein geratener Maßstab verschöbe jeden Rahmen um denselben Faktor — gleichmäßig und damit unauffällig falsch. Ein Rahmen an der falschen Höheneinheit ist schlimmer als kein Rahmen.

---

8 Übernahme aus einer NetBox-Instanz

> Dieser Abschnitt ist der **einzige** im Handbuch, der ein Fremdprodukt beim Namen nennt — und das aus gutem Grund: Sie müssen wissen, woraus Sie übernehmen.

Wenn Sie die physische Dokumentation heute in **NetBox** führen, übernimmt Ordivis Platform diesen Bestand. Sie finden die Funktion in der Administration unter **CMDB → Übernahme**.

Übernommen werden:

Aus NetBox	Wird in Ordivis Platform zu
Sites	Standort-CIs
Locations	Raum-CIs, in die Standortkette eingehängt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Aus NetBox	Wird in Ordivis Platform zu
Racks	Rack-CIs samt Höhe, unterster Einheit und Zählrichtung
Device Types	Einträge im Gerätemodell-Katalog
Devices	CIs der zugeordneten Klasse, samt Einbauposition
Interfaces, Console-, Power-, Front- und Rear-Ports	Geräte-Komponenten
Cables	Logische Verbindungen zwischen Anschlüssen

8.1 Es ist eine Migration, keine Dauersynchronisation

Das ist eine bewusste Festlegung und keine technische Einschränkung. Eine wiederkehrende Abgleichstrecke machte die Fremdquelle zur **zweiten Wahrheit** neben Ihrer CMDB — genau der Zustand, gegen den eine CMDB antritt. Deshalb gilt:

- Es gibt **keinen Zeitplan, keinen Auslöser und keinen Webhook**. Die Übernahme wird von Hand gestartet.
- Der Lauf ist **innerhalb Ihres Migrationsprojekts beliebig oft wiederholbar**. Übernommene Objekte werden wiedererkannt; ein zweiter Anlauf verdoppelt nichts. Sie brauchen mehrere Anläufe, bis die Zuordnung sitzt — dafür ist das gedacht.
- Nach der Übernahme **schalten Sie die Verbindung ab**. Der Zugriffsschlüssel wird dabei gelöscht, und der Zustand ist im Produkt sichtbar. Ein Schlüssel, der danach liegen bleibt, wäre ein dauerhaft gültiger Fremdzugang ohne Zweck.

8.2 Zuordnung der CI-Klassen

NetBox kennt Geräterollen und Gerätetypen; Ordivis Platform kennt CI-Klassen. Die Zuordnung ist eine **pflegbare Tabelle** mit Vorbelegung, und der Gerätetyp sticht die Rolle: Die Rolle beschreibt den Einsatzzweck, der Typ das Gerät.

Was sich nicht zuordnen lässt, wird berichtet und nicht geraten. Ein falsch geratener CI-Typ ist in einer CMDB teurer als ein fehlender: Der fehlende fällt auf und wird nachgepflegt, der falsche steht jahrelang plausibel da und verfälscht jede Auswertung, die auf der Klasse aufsetzt.

8.3 Vorgehen

1. **Verbindung hinterlegen** — Adresse der Instanz und einen Zugriffsschlüssel mit Leserechten auf den DCIM-Bereich. Der Schlüssel wird geschützt abgelegt und nie im Klartext gespeichert.
2. **Verbindung prüfen.** Ordivis Platform meldet die Fassung der Quellinstanz. Ist es nicht die unterstützte Reihe, bricht der Lauf mit klarer Meldung ab — eine fremde Schnittstellenfassung zu erraten importierte stillschweigend Unsinn.
3. **Probelauf** über die Stammdaten. Der Bericht weist jede Zeile einzeln aus.
4. **Zuordnungstabelle prüfen** und ergänzen. Das ist der Schritt, für den es den Probelauf gibt.
5. **Stammdaten übernehmen**, dann Geräte, Komponenten und Verbindungen.
6. **Verbindung abschalten.**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8.4 Die Falle bei der Einbauposition

NetBox erlaubt **halbe Höheneinheiten** als Dezimalwert; Ordivis Platform rechnet im Belegungsplan in ganzen Einheiten. Die Umrechnung erfolgt durchgehend in halben Einheiten und bildet erst am Ende auf ganze ab.

Zwei halbhohe Geräte, die sich in der Quelle eine Höheneinheit teilen, belegen hier beide dieselbe ganze Einheit. Das zweite wird deshalb als **Kollision berichtet** und nicht stillschweigend eine Einheit höher gesetzt, wo es nicht steht. Dasselbe gilt für jede andere Kollision: Die Einbauposition bleibt dann leer, das Gerät kommt trotzdem herein, und der Abschlussbericht nennt den Fall.

---

9 Rollen & Rechte (Auszug)

Aufgabe	Benötigtes Recht
Katalog und Komponenten ansehen, Weg verfolgen	`cmdb.ci.read`
Komponenten pflegen, Einbauten und Vormerkungen ändern	`cmdb.ci.update`
Gerätemodelle und Rackmodelle pflegen	`cmdb.class.manage`
Modelldateien einlesen, Übernahme aus einer Fremdinstanz	`cmdb.import`

Verwandte Themen

- **Modul CMDB** — CIs, Klassen, Beziehungen, Standortbaum
- **Modul IPAM** — IP-Adressen an Netzwerkschnittstellen
- **Modul Asset- und Vertragsmanagement** — kaufmännische Sicht auf dieselben Geräte
- **Datenimport in Prozesse** — Massenimport von CIs

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · IPAM

Modul IPAM (IP-Adressverwaltung)

IP-Adressverwaltung – Subnetze, Adressen, DHCP, Netzwerk-Scans

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-22 IPAM

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul IPAM (IP-Adressverwaltung)

Jedes Gerät in einem Netzwerk braucht eine eigene Adresse, damit die Daten beim richtigen Empfänger ankommen – ähnlich wie jedes Haus eine Postanschrift hat. Diese Adressen heißen IP-Adressen, zum Beispiel `10.10.0.15`. In einem kleinen Netz behält man sie noch im Kopf oder auf einem Zettel. Sobald aber Hunderte Geräte dazukommen, wird daraus schnell ein Durcheinander: Zwei Geräte bekommen versehentlich dieselbe Adresse, niemand weiß mehr, welche noch frei ist, und ein Ausfall lässt sich nur mühsam einer Adresse zuordnen.

IPAM steht für *IP Address Management*, die Verwaltung von IP-Adressen. Das Modul führt Buch darüber, welche Netze und Teilnetze (Subnetze) es gibt, welche Adressen belegt, frei oder reserviert sind und zu welchem Gerät eine Adresse gehört. Über die Netzwerkschnittstellen der CMDB (ID-21) ist jede Adresse mit dem zugehörigen Configuration Item verknüpft, sodass Sie an einer Stelle sehen, was wohin gehört.

INHALT

1 Zweck & Nutzen
2 Ein Subnetz anlegen
3 IP-Adressen verwalten
3.1 Eine Adresse reservieren oder freigeben
3.2 Mehrere Adressen und ganze Bereiche an einer Schnittstelle
4 DHCP-Reservierungen
4.1 Kennzeichnung des DHCP-Bereichs
5 Netzwerk-Scans
5.1 Daten von außen: entfernte Netze und Zulieferer
6 IPAM-Dashboard
7 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Überblick über belegte, freie und reservierte Adressen.
- Vermeidung von IP-Konflikten.
- Verknüpfung von IP-Adressen mit CIs (über deren Schnittstellen).

2 Ein Subnetz anlegen

💻 **Nur im Client:** *Netz (IPAM)* ▸ *IP-Adressen* ▸ *Subnetz hinzufügen* · 🔑 `ipam.prefix.manage` — Das Portal zeigt Präfixe an und verwaltet Adressen, legt aber keine Subnetze an.

🔒 **Pflichtreihenfolge** — die Adressliste entsteht erst aus Netzadresse und Präfix. Alles Weitere (VLAN, Standort, DHCP-Bereich) hängt am gespeicherten Subnetz.

1. Öffnen Sie *Netz (IPAM)* ▸ *IP-Adressen*.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2. Klicken Sie auf *Subnetz hinzufügen*.
3. Geben Sie **Netzadresse und Präfix** an (z. B. `10.10.0.0/24`).
4. Ergänzen Sie bei Bedarf Beschreibung, VLAN und Standort.
5. Speichern Sie.

✅ **Ergebnis:** Das Subnetz steht im Prefix-Baum links, mit Belegungsübersicht und der vollständigen Adressliste rechts.

3 IP-Adressen verwalten

> 💻 **Client:** *Netz (IPAM)* ▸ *IP-Adressen* · 🌐 **Portal:** *Netz (IPAM)* ▸ *Adressverwaltung* · 🔑 `ipam.prefix.read` zum Lesen, `ipam.ip.assign` zum Zuweisen

🔒 **Pflichtreihenfolge** — die Adressliste rechts entsteht erst durch die Auswahl links. Steht im Portal *Kein Präfix gewählt* mit dem Hinweis *Links ein Präfix wählen, um seine Adressen zu sehen.*, fehlt genau dieser erste Klick.

1. Wählen Sie links im Prefix-Baum ein Subnetz aus.
2. Lesen Sie rechts die Adressliste. Die Spalte *Zugewiesen* zeigt immer den **Klarnamen** des verknüpften CI, nie eine ID.

✅ **Ergebnis:** Sie sehen jede Adresse des Subnetzes mit Status, Hostname und zugewiesenem CI.

Abbildung 1 zeigt beide Bereiche nebeneinander.

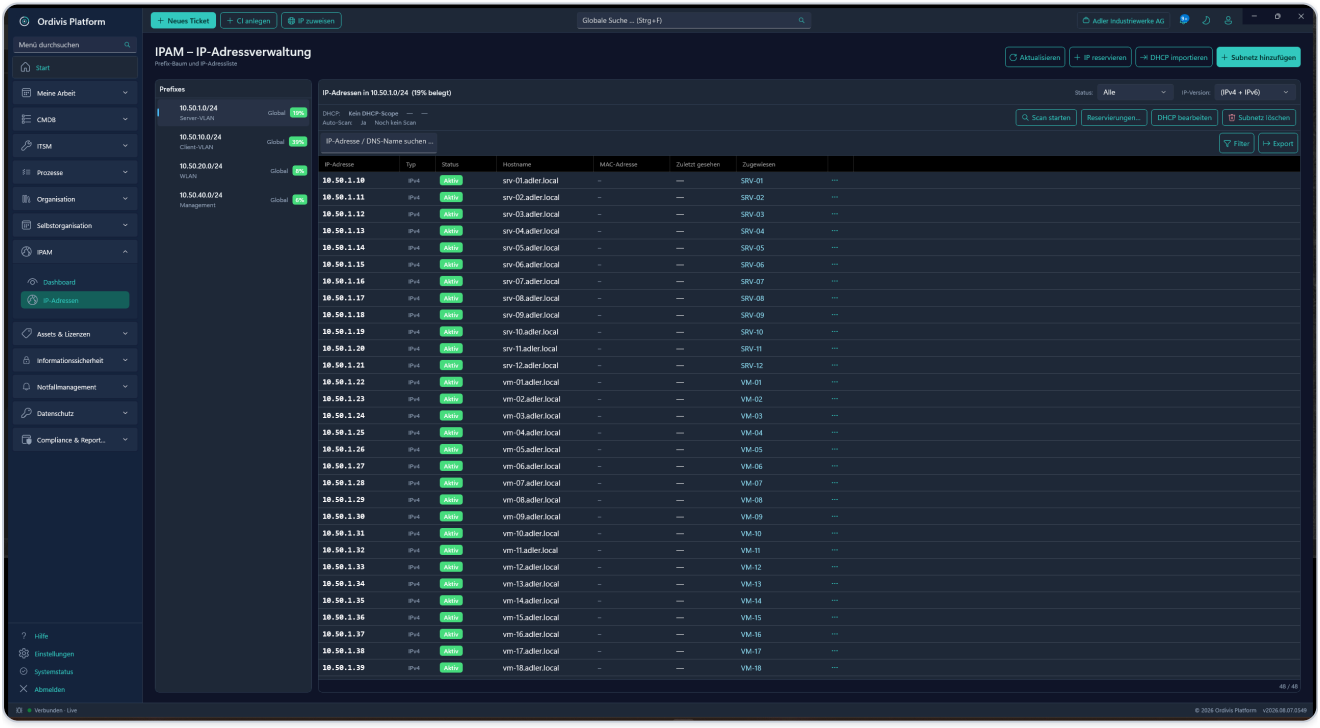

Abb. 1 – IP-Adressverwaltung: Prefix-Baum links, Adressliste mit Status, Hostname und zugewiesenem CI rechts.

Innerhalb eines Subnetzes sehen Sie alle Adressen mit **Status**:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Status	Bedeutung
Frei	verfügbar
Belegt	einem CI/Host zugeordnet
Reserviert	fest vergeben (z. B. Gateway, Drucker)

3.1 Eine Adresse reservieren oder freigeben

> 💻 **Client:** *Netz (IPAM)* ▸ *IP-Adressen* ▸ *IP reservieren* · 🌐 **Portal:** *Netz (IPAM)* ▸ *Adressverwaltung* ▸ *Reservieren* · 🔑 `ipam.ip.assign`

So reservieren Sie eine Adresse:

1. Wählen Sie links das Präfix aus.
2. Klicken Sie auf *Reservieren* (im Client *IP reservieren*).
3. Tragen Sie die *Adresse* ein.
4. Halten Sie unter *Grund* fest, wofür sie reserviert ist – „Gateway“, „Drucker 2. OG“. Ohne Grund weiß in einem Jahr niemand mehr, warum die Adresse blockiert ist.
5. Bestätigen Sie.

✅ **Ergebnis:** Es erscheint *Adresse reserviert.* Die Adresse steht auf *Reserviert* und wird nicht mehr automatisch vergeben.

So geben Sie eine Adresse wieder frei: Klicken Sie in der Zeile auf *Freigeben*. Es erscheint *Adresse freigegeben.*, und der Status wechselt auf *Frei*.

Eine Adresse einem Gerät zuordnen: Das geschieht nicht hier, sondern **an der Schnittstelle des CI** (Abschnitt 3.2). So bleibt die IP-Information an einer Stelle; die CMDB zeigt sie nur lesend an.

> **Hinweis:** In Listen wird immer der **Klarname** des zugeordneten CI angezeigt, nie eine ID.

> **Hinweis:** Die Spalte *Herkunft* im Portal unterscheidet *Gepflegt* von *Nur gescannt* (*Aus einem Scan, noch nicht bestätigt*). Eine gescannte Adresse ist eine Beobachtung, keine Zusage – bestätigen Sie sie, bevor Sie sich darauf verlassen (`ipam.ip.confirm`).

3.2 Mehrere Adressen und ganze Bereiche an einer Schnittstelle

Eine Schnittstelle kann **mehrere Adressen** tragen. Sie erscheinen alle in der Schnittstellenliste des CI; bei mehr als zwei zeigt die Zelle die ersten beiden und einen Zähler.

Für den Fall, dass ein Anbieter einen **ganzen Block** auf einen Anschluss legt — ein öffentliches `/29` am WAN-Port einer Firewall —, ordnen Sie den **Bereich als Ganzes** zu:

> 💻 **Nur im Client:** *CMDB* ▸ *Configuration Items* ▸ *CI öffnen* ▸ *Schnittstellen* · 🔑 `ipam.ip.assign`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Öffnen Sie das CI in der CMDB.
2. Wechseln Sie auf den Bereich *Schnittstellen*.
3. Klicken Sie in der Zeile der Schnittstelle auf … und wählen Sie *Bereich zuweisen…*.
4. Wählen Sie den Bereich aus der Liste.
5. Bestätigen Sie.

✅ **Ergebnis:** Der Bereich hängt an der Schnittstelle und erscheint in der Schnittstellenliste des CI.

Angeboten werden nur Bereiche, die an keiner Schnittstelle hängen. Ein Bereich gehört zu genau einer Schnittstelle; eine Schnittstelle kann mehrere Bereiche tragen (etwa einen v4-Block und ein v6-Präfix).

> **Warum nicht acht Einzeladressen.** Ein `/29` einzeln zuzuordnen hieße, acht Adressen zu pflegen und den Zusammenhang „das ist ein Block" zu verlieren. Beim nächsten Anschlusswechsel müsste man ihn aus acht Zeilen wieder zusammensuchen.

Zuordnung aufheben löst Adresse oder Bereich von der Schnittstelle; beide bleiben bestehen. Wird eine Schnittstelle **gelöscht**, werden ihre Adressen und Bereiche automatisch frei — ein Bereich, der auf eine verschwundene Schnittstelle zeigt, erschiene in keiner Liste mehr und wäre dauerhaft blockiert.

4 DHCP-Reservierungen

> 💻 **Nur im Client:** *Netz (IPAM)* ▸ *IP-Adressen* ▸ *Reservierungen…* bzw. *DHCP importieren* · 🔑 `ipam.ip.assign`, für den Import `ipam.dhcp.import`

Sie pflegen **DHCP-Reservierungen** (MAC → feste IP) und können **Windows-DHCP** importieren, um bestehende Reservierungen zu übernehmen. So spiegelt IPAM die reale DHCP-Konfiguration.

So übernehmen Sie bestehende Reservierungen aus Windows-DHCP:

🔒 **Pflichtreihenfolge** — das Subnetz muss angelegt sein (Abschnitt 2), sonst haben die importierten Reservierungen kein Zuhause.

1. Öffnen Sie *Netz (IPAM)* ▸ *IP-Adressen*.
2. Klicken Sie auf *DHCP importieren*.
3. Wählen Sie die Quelle aus und starten Sie den Lauf.
4. Prüfen Sie die **Vorschau** – der Import zeigt vor der Übernahme, was er anlegen würde.
5. Bestätigen Sie die Übernahme.

✅ **Ergebnis:** Subnetze und Reservierungen stehen in der IPAM. Über *DHCP bearbeiten* am Subnetz pflegen Sie den Bereich danach weiter.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4.1 Kennzeichnung des DHCP-Bereichs

> 💻 **Nur im Client:** *Netz (IPAM)* ▸ *IP-Adressen* ▸ *DHCP bearbeiten* am Subnetz, mandantenweit unter *Fußbereich* ▸ *Einstellungen* ▸ *IPAM* · 🔑 `ipam.prefix.manage`

Ist am Subnetz ein DHCP-Bereich hinterlegt, werden alle Adressen darin in der Adressliste mit **DHCP** gekennzeichnet. Die Kennzeichnung wird aus dem Bereich **berechnet** und jedes Mal frisch ermittelt – der gespeicherte Status einer Adresse bleibt davon unberührt.

🔒 **Pflichtreihenfolge** — die Kennzeichnung braucht **Start und Ende**. Mit nur einer der beiden Grenzen passiert nichts, und zwar wortlos.

1. Öffnen Sie am Subnetz *DHCP bearbeiten*.
2. Tragen Sie die **Startadresse** des DHCP-Bereichs ein.
3. Tragen Sie die **Endadresse** ein.
4. Speichern Sie.

✅ **Ergebnis:** Alle Adressen zwischen beiden Grenzen tragen in der Liste die Marke *DHCP*.

Mandantenweit steuern Sie die Anzeige unter *Einstellungen* ▸ *IPAM* ▸ *DHCP-Bereiche im Subnetz kennzeichnen*. Die Einstellung ist standardmäßig eingeschaltet; ausschalten blendet die Kennzeichnung lediglich wieder aus und ändert keine Daten.

> **Hinweis:** Ist der Bereich unvollständig (nur Start oder nur Ende) oder liegen die Grenzen verdreht, wird bewusst **nichts** gekennzeichnet – eine falsche Kennzeichnung wäre schlechter als gar keine.

5 Netzwerk-Scans

> 💻 **Nur im Client:** *Netz (IPAM)* ▸ *IP-Adressen* ▸ *Scan starten* · 🔑 `ipam.prefix.manage` — Das Portal zeigt die Ergebnisse in der Spalte *Herkunft*, startet aber keinen Scan.

Über **Netzwerk-Scans** erfassen Sie den Ist-Zustand eines Subnetzes (welche Adressen antworten). Treffer können mit vorhandenen CIs abgeglichen werden. Tiefergehende, authentifizierte Scans gehören zum Discovery-Modul (ID-24).

🔒 **Pflichtreihenfolge** — der Scan läuft gegen **ein** Subnetz. Ohne Auswahl im Prefix-Baum weiß er nicht, wo er suchen soll.

1. Wählen Sie links im Prefix-Baum das Subnetz aus.
2. Klicken Sie auf *Scan starten*.
3. Warten Sie das Ende des Laufs ab.
4. Prüfen Sie die neuen Adressen: Sie stehen als **unbestätigt** in der Liste (*Nur gescannt*).
5. Bestätigen Sie, was stimmt (`ipam.ip.confirm`).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

✅ **Ergebnis:** Der Ist-Zustand des Netzes steht in der Adressliste, deutlich getrennt nach gepflegt und gescannt.

Achtung: Ein Scan sieht nur, was zum Zeitpunkt des Laufs geantwortet hat. Ein ausgeschaltetes Gerät erscheint als freie Adresse. Vergeben Sie eine Adresse nie allein deshalb neu, weil ein Scan sie nicht gefunden hat.

Tipp: Über *Auto-Scan:* am Subnetz stellen Sie einen wiederkehrenden Lauf ein. Das ist dem Klicken von Hand vorzuziehen – ein Scan, an den man denken muss, findet irgendwann nicht mehr statt.

5.1 Daten von außen: entfernte Netze und Zulieferer

Ein IPAM ist nur so gut wie seine Abdeckung – und die endet dort, wo der Server nicht hinkommt. Für entfernte Segmente gibt es zwei Wege, die dieselbe Regel befolgen: **Eingespeistes wird nie blind übernommen.**

- **Discovery-Collector** (ID-24, Abschnitt 6): ein Dienst im entfernten Netz, dauerhaft und mit Zeitplan.
- **Externe Werkzeuge** wie das kostenlose **Ordivis Toolbox**: laufen am Arbeitsplatz einer Administratorin und liefern auf Zuruf.

Was dabei ankommt und wie es aussieht:

Zulieferung	Ergebnis in der IPAM
Netzwerkscan	neue Adressen als **unbestätigt**, neues Subnetz im Status **„Reserviert"**
DHCP-Bereiche und Reservierungen	Subnetze und Reservierungen (mit Vorschau vor der Übernahme)
DNS-/PTR-Abgleich	setzt den **DNS-Namen** an bereits bekannten Adressen
AD Sites & Services	ordnet einem Subnetz den **Standort** zu

Zum DNS-Abgleich. Er benennt, er legt nicht an: Nur Adressen, die die IPAM schon kennt, bekommen einen Namen. Widersprüche zwischen Vorwärts- und Rückwärtsauflösung und Einträge, die auf eine unbekannte Adresse zeigen, werden **gemeldet** – nicht bereinigt. Einen DNS-Eintrag zu löschen ist eine Entscheidung mit Folgen; sie gehört nicht in einen Abgleichlauf.

Zum Standort. Die Angabe aus AD Sites & Services ist autoritativ und wird nur dann verknüpft, wenn es in der CMDB ein **Standort-CI dieses Namens** gibt. Fehlt es, reist der Name als Hinweis im Empfangsbeleg mit – Sie sehen dann, welcher Standort in der CMDB noch fehlt. Eine bereits vorhandene Zuordnung wird **nie** überschrieben: Die Entscheidung eines Menschen wird nicht durch einen Scan ersetzt.

Die Einrichtung des Maschinen-Zugangs beschreibt das Betriebshandbuch (**ID-50**, Abschnitt 7.4.1).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6 IPAM-Dashboard

> 💻 **Nur im Client:** *Netz (IPAM)* ▸ *Dashboard* · 🔑 `ipam.prefix.read` — Das Portal hat kein eigenes IPAM-Dashboard; die Belegung je Präfix steht dort über der Adressliste (*{belegt} von {gesamt} Adressen belegt*).

Das Dashboard zeigt Auslastung je Subnetz, Konflikte und Trends – hilfreich für die Kapazitätsplanung. Sehen Sie dort turnusmäßig nach: Ein Subnetz, das die 90 Prozent überschreitet, ist kein Problem von heute, sondern eines vom nächsten Rollout.

7 Rollen & Rechte (Auszug)

Recht	Erlaubt
`ipam.prefix.read` / `ipam.ip.read`	Subnetze/Adressen ansehen
`ipam.prefix.manage` / `ipam.ip.assign`	Subnetze/Adressen pflegen und zuweisen
`ipam.ip.confirm`	unbestätigte Scan-Treffer ratifizieren
`ipam.dhcp.import`	DHCP-Bereiche importieren

> **Tipp:** Reservieren Sie Infrastruktur-Adressen (Gateway, DNS, Drucker) bewusst als „Reserviert“, damit sie nicht versehentlich vergeben werden.

Verwandte Themen

- **ID-21 – Modul CMDB** – Verknüpfung von Adresse und Configuration Item
- **ID-24 – Modul Discovery** – Netzbereiche automatisch abtasten
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Ports und Freigaben im Serverbetrieb

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · ASSETS & VERTRÄGE

Modul Asset- & Vertragsmanagement

Geräte, Verträge und Software-Lizenzen über den Lebenszyklus

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-23 Assets

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Asset- & Vertragsmanagement

Ein Notebook ist mehr als ein Gerät im Netzwerk. Es wurde gekauft und hat einen Preis, es gehört zu einem Vertrag, es hat eine Garantie, die irgendwann ausläuft, und auf ihm laufen Programme, die lizenziert sein müssen. Diese kaufmännische Seite der IT gerät im Alltag leicht aus dem Blick, bis eine Garantie ungenutzt verfällt, ein Wartungsvertrag sich stillschweigend verlängert oder bei einer Prüfung mehr Software installiert ist als lizenziert.

„Asset Management" heißt übersetzt die Verwaltung von Vermögenswerten. Ein *Asset* ist in diesem Sinn ein Wert, den die Organisation besitzt oder für den sie zahlt: ein Gerät, ein Vertrag, eine Lizenz. Dieses Modul begleitet solche Werte über ihren gesamten Lebensweg, von der Beschaffung über den Betrieb bis zur Stilllegung, und behält dabei Kosten, Fristen und Lizenz-Compliance im Auge. Es beantwortet nicht die Frage „Wie läuft das Gerät?", sondern „Was kostet es uns, was schulden wir vertraglich, und sind wir sauber lizenziert?".

INHALT

1 Abgrenzung zur CMDB
2 Assets (Cradle-to-Grave)
2.1 Ein Asset anlegen und über seinen Lebensweg führen
3 Verträge
3.1 Einen Lieferanten anlegen
3.2 Einen Vertrag anlegen
4 Software-Lizenzen (kundeneigen)
5 Erkannte Software & Lizenzzählung
6 Kosten-Roll-up & Governance
7 Rollen & Rechte (Auszug)
Verwandte Themen

1 Abgrenzung zur CMDB

- **Asset Management:** kaufmännische Sicht – Kosten, Verträge, Garantie, Lizenzen.
- **CMDB/CI:** betriebliche Sicht – Relationen, Impact, Betrieb.

Dasselbe Objekt (z. B. ein Server) kann **beides** sein. Beide Sichten werden **verknüpft**, nicht doppelt gepflegt.

2 Assets (Cradle-to-Grave)

Die Asset-Liste führt alle Vermögenswerte mit Status, Typ, Seriennummer, Garantie und Kaufpreis – die kaufmännische Gegensicht zur CI-Liste der CMDB.

Abbildung 1 zeigt die Liste mit Garantie-Ablauf und Kaufpreis.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

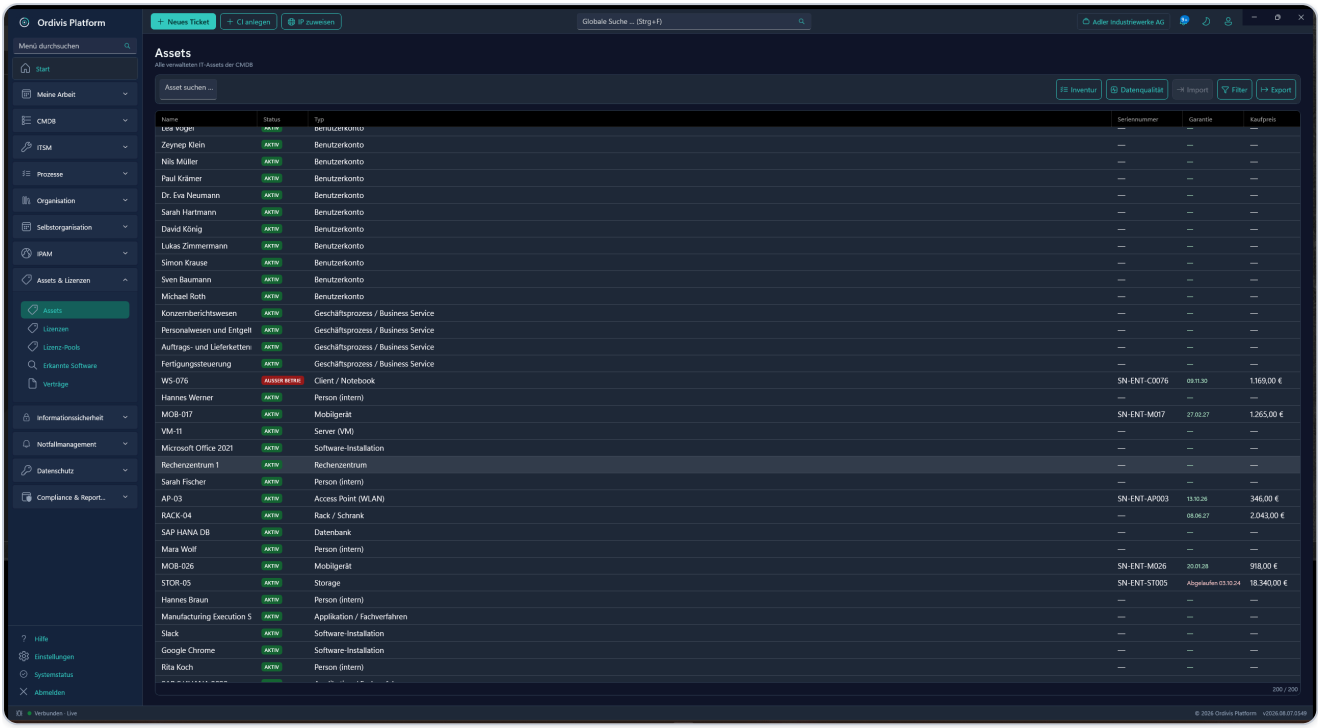

Abb. 1 – Asset-Liste: Garantie-Ablauf (inkl. Warnung „Abgelaufen“) und Kaufpreis je Vermögenswert.

Ein Asset durchläuft den Lebenszyklus **Beschaffung → Betrieb → Stilllegung**.

2.1 Ein Asset anlegen und über seinen Lebensweg führen

> 💻 **Client:** *Assets & Lizenzen ▸ Assets* · 🌐 **Portal:** *Assets & Lizenzen ▸ Assets* · 🔑 `cmdb.ci.update`, für den Statuswechsel zusätzlich `cmdb.ci.lifecycle.manage`

💡 **Empfohlene Reihenfolge** — hinterlegen Sie die Garantie beim Anlegen, nicht später. Nachträglich gepflegte Garantiedaten sind der häufigste Grund dafür, dass eine Frist unbemerkt verstreicht.

1. **Anlegen:** Klicken Sie auf *Neues Asset* und erfassen Sie Typ, Bezeichnung, Inventarnummer und Kosten.
2. **Garantie/Gewährleistung:** Hinterlegen Sie das Ablaufdatum. Es erscheint danach in Warnungen und Berichten.
3. **Zuordnung:** Ordnen Sie das Asset einer Person, einem Standort oder einem verknüpften CI zu.
4. **Stilllegung:** Setzen Sie am Ende den Status. Die historischen Daten bleiben erhalten.

✅ **Ergebnis:** Das Asset steht in der Liste (Abbildung 1) mit Garantie-Ablauf und Kaufpreis; ein abgelaufener Schutz ist mit *Abgelaufen* gekennzeichnet.

> **Achtung:** Ein stillgelegtes Asset wird **nicht gelöscht**. Das ist gewollt – die Kosten der Vergangenheit gehören in jede Auswertung über mehrere Jahre. Löschen entzieht der Auswertung die Grundlage.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3 Verträge

> 💻 **Client:** *Assets & Lizenzen* ▸ *Verträge* · 🌐 **Portal:** *Assets & Lizenzen* ▸ *Verträge* · 🔑 `contracts.read` zum Lesen, `contracts.manage` zum Pflegen

Verträge bündeln vertragliche Verpflichtungen: Lieferanten-Stammdaten, Kosten (mit Roll-up in die Berichte), Kündigungs- und Verlängerungsfristen sowie die Governance-Angaben – Verantwortliche, Erinnerungen, Statusübersicht.

3.1 Einen Lieferanten anlegen

> 🌐 **Nur im Portal:** *Assets & Lizenzen* ▸ *Lieferanten* ▸ *Neuer Lieferant* · 🔑 `contracts.manage` — Im Client erreichen Sie die Lieferanten über *Assets & Lizenzen* ▸ *Verträge*, Schaltfläche *Lieferanten* in der Werkzeugleiste.

1. Klicken Sie auf *Neuer Lieferant*.
2. Tragen Sie *Name*, *E-Mail* und *Telefon* ein.
3. Klicken Sie auf *Speichern*.

✅ **Ergebnis:** Es erscheint *Lieferant gespeichert.* Der Lieferant steht ab jetzt bei jedem Vertrag zur Auswahl.

3.2 Einen Vertrag anlegen

> 💻 **Nur im Client:** *Assets & Lizenzen* ▸ *Verträge* ▸ *Neuer Vertrag* · 🔑 `contracts.manage` — Das Portal zeigt Verträge mit *Vertrag*, *Lieferant*, *Status* und *Ende* an; angelegt werden sie im Client.

> **Voraussetzung:** Der Lieferant ist angelegt (Abschnitt 3.1).

🔒 **Pflichtreihenfolge** — ohne Lieferant lässt sich der Vertrag nicht zuordnen, und ohne Vertragsende rechnet die Kündigungsfrist gegen nichts.

1. Klicken Sie auf *Neuer Vertrag*.
2. Tragen Sie *Titel* ein und wählen Sie den *Lieferanten*.
3. Wählen Sie den *Typ* (Wartung, Support, Lizenz …).
4. Setzen Sie *Vertragsende*.
5. Tragen Sie die *Kündigungs*frist ein – die Zeitspanne, die vor dem Ende einzuhalten ist.
6. Erfassen Sie die *Jahreskosten*.
7. Hinterlegen Sie unter *Governance* die verantwortliche Person.
8. Speichern Sie.

✅ **Ergebnis:** Der Vertrag steht in der Liste. Über *Ablaufende Verträge* sehen Sie jederzeit, was als Nächstes ansteht.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Tipp:** Hinterlegen Sie Kündigungsfristen sorgfältig – Ordivis Platform warnt rechtzeitig und verhindert ungewollte automatische Verlängerungen. Die Erinnerung selbst richten Sie in ID-31 ein; ohne sie steht die Frist zwar da, meldet sich aber nicht.

4 Software-Lizenzen (kundeneigen)

> 💻 **Client:** *Assets & Lizenzen* ▸ *Lizenzen* und *Lizenz-Pools* · 🌐 **Portal:** *Assets & Lizenzen* ▸ *Lizenzen* · 🔑 `licensing.read` zum Lesen, `licensing.manage` zum Pflegen — **Lizenz-Pools** und **Erkannte Software** gibt es nur im Client.

Hier verwalten Sie die **eigenen Software-Lizenzen** der Organisation:

Abbildung 2 zeigt die Lizenzsicht mit Metrik, Menge und Ablaufdatum.

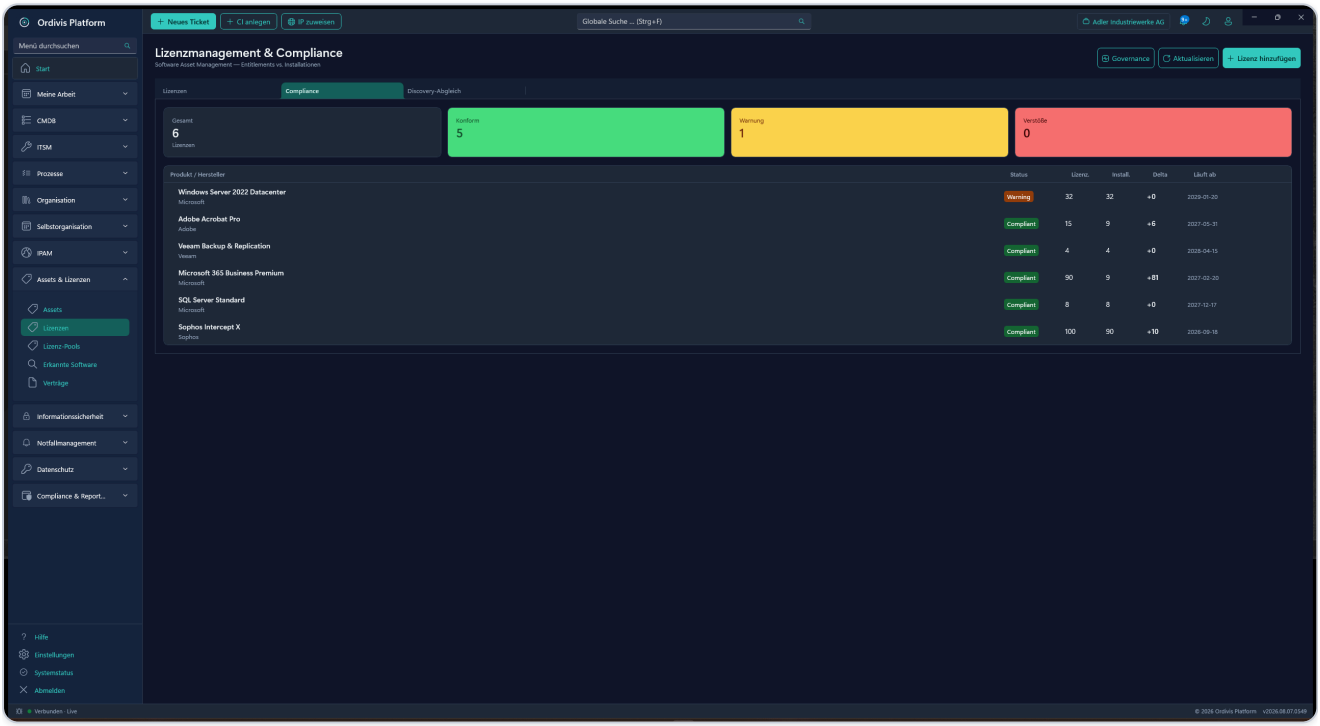

Abb. 2 – Lizenzmanagement & Compliance: Lizenzmetrik (PerUser, PerCore, SaaS …), Menge und Ablaufdatum je Produkt.

- **Lizenz-Bestand** (Backlog): erworbene Lizenzen mit Menge und Metriken.
- **Lizenzpools:** bündeln Lizenzen einer Software zur Zuteilung.
- **Compliance:** Abgleich von **erworbenen** gegen **installierte** Lizenzen.

> **Wichtig:** Diese Software-Lizenzen sind **kundeneigen** und nicht mit der **Produktlizenz von Ordivis Platform** (ID-43) zu verwechseln.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5 Erkannte Software & Lizenzzählung

> 💻 **Nur im Client:** *Assets & Lizenzen* ▸ *Erkannte Software* · 🔑 `licensing.read` — Im Portal gibt es diese Sicht nicht.

Per Discovery inventarisierte Anwendungen (Hersteller, Name, SKU, Version) speisen die Compliance-Rechnung. Gezählt wird je **Hauptversion**: Builds und Unterversionen kollabieren auf die lizenzrelevante Edition. So entspricht die Zählung dem, was tatsächlich lizenziert werden muss.

So prüfen Sie die Lizenz-Compliance:

🔒 **Pflichtreihenfolge** — die Rechnung stimmt nur, wenn beide Seiten aktuell sind. Ein Discovery-Lauf vor dem Abgleich ist keine Formsache, sondern die Voraussetzung dafür, dass das Ergebnis etwas wert ist.

1. Stoßen Sie einen Discovery-Lauf an oder prüfen Sie, wann der letzte war (ID-24).
2. Öffnen Sie *Erkannte Software* und sehen Sie die installierten Anwendungen durch.
3. Öffnen Sie *Lizenzen* und prüfen Sie den erworbenen Bestand je Produkt.
4. Vergleichen Sie erworbene gegen installierte Lizenzen.

✅ **Ergebnis:** Sie kennen je Produkt die Über- oder Unterdeckung – und zwar belegbar, nicht geschätzt.

> **Achtung:** Discovery findet, was installiert ist, nicht was **genutzt** wird. Eine Software auf einem seit Monaten abgeschalteten Rechner zählt in dieser Rechnung mit. Prüfen Sie bei einer Unterdeckung zuerst, ob die Installationen alle noch aktiv sind.

6 Kosten-Roll-up & Governance

Die Detailseite eines Assets bündelt die kaufmännische 360°-Sicht: Beschaffung, TCO und Abschreibung, Garantie sowie die Bezüge zu CMDB, Verträgen und Tickets.

Abbildung 3 zeigt eine solche Detailseite.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

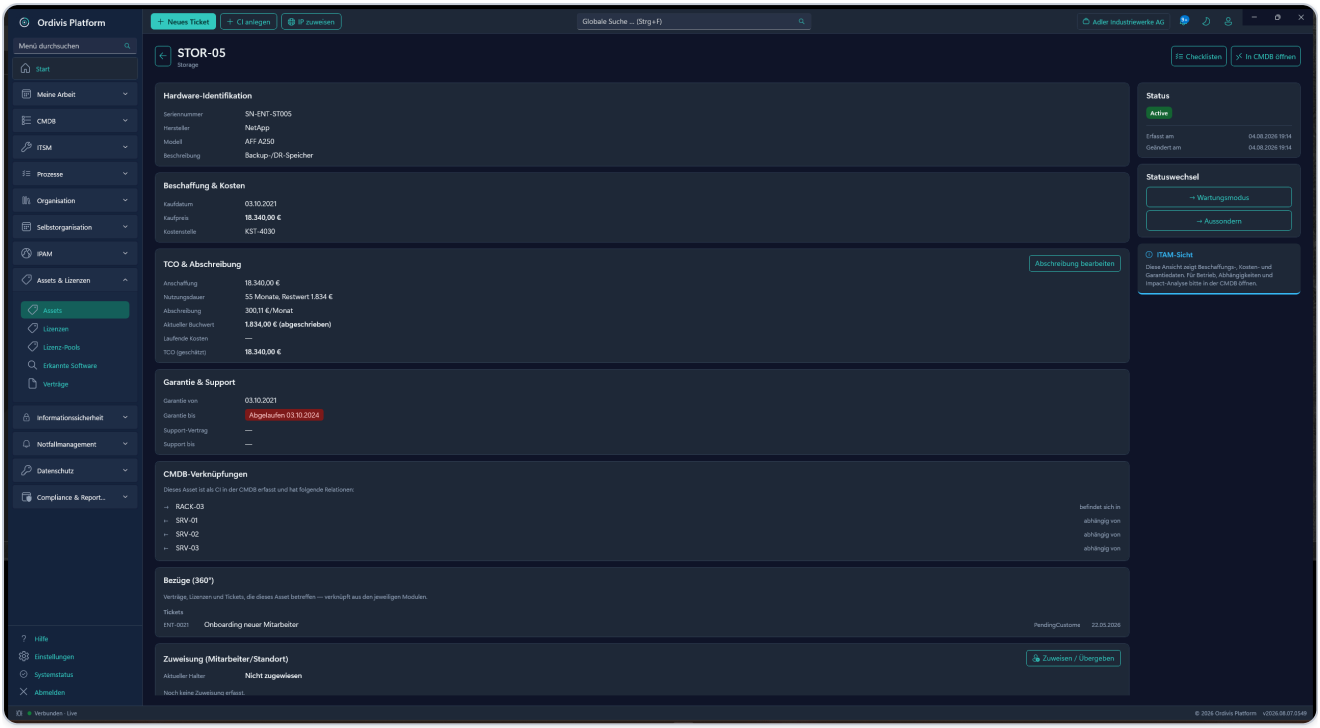

Abb. 3 – Asset-Detail: Beschaffung & Kosten, TCO/Abschreibung (Buchwert), Garantie und Verknüpfungen zur CMDB.

Kosten aus Assets und Verträgen werden zu Summen je Kategorie/Kostenstelle aggregiert und stehen in Reporting/Dashboards (ID-27) bereit. Governance-Funktionen (Verantwortliche, Fristen, Freigaben) sorgen für revisionssichere Nachvollziehbarkeit.

7 Rollen & Rechte (Auszug)

Recht	Erlaubt
`cmdb.ci.read`	Assets ansehen (Assets werden als CI geführt)
`cmdb.ci.update` / `cmdb.ci.lifecycle.manage`	Assets pflegen, Lebenszyklus setzen
`contracts.read` / `contracts.manage`	Verträge und Lieferanten ansehen / verwalten
`licensing.read` / `licensing.manage`	Lizenzdaten ansehen / verwalten

Verwandte Themen

- **ID-21 – Modul CMDB** – warum Geräte als CI geführt werden
- **ID-43 – Modul Produktlizenzierung** – Abgrenzung zur Lizenzierung von Ordivis selbst
- **ID-31 – Modul Benachrichtigungen & E-Mail** – Erinnerung an Vertrags- und Garantiefristen
- **ID-27 – Modul Reporting & Dashboards** – Kosten und Laufzeiten auswerten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · DISCOVERY

Modul Discovery (Netzwerk-Inventarisierung)

Automatische Netzwerk-Inventarisierung per WMI, SSH, SNMP und LDAP

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-24 Discovery

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Discovery (Netzwerk-Inventarisierung)

Eine CMDB ist nur so nützlich wie ihre Aktualität. Pflegt man jeden Server und jeden PC von Hand ein, hinkt der Datenbestand der Wirklichkeit fast immer hinterher: Geräte kommen dazu, verschwinden oder bekommen neue Software, ohne dass es jemand nachträgt. Auf Dauer traut niemand mehr den Einträgen.

Discovery nimmt Ihnen diese Fleißarbeit ab. „Discovery“ heißt Erkennung: Das Modul durchsucht das Netzwerk selbstständig und findet heraus, welche Geräte es gibt und was auf ihnen läuft. Ein einfacher Netzwerk-Scan prüft, welche Adressen überhaupt antworten. Der ausführlichere Deep-Scan meldet sich mit hinterlegten Zugangsdaten an einem System an und liest Details aus, bis hin zur installierten Software. Die Funde landen nicht ungefragt in der CMDB, sondern zuerst als Vorschläge (Kandidaten), die Sie prüfen und übernehmen. So bleibt die Dokumentation aktuell, ohne dass Sie die Kontrolle abgeben.

INHALT

1 Zweck & Nutzen
2 Scan-Arten
3 Anmeldedaten (Credential-Vault)
4 Einen Scan durchführen
5 Scan-Abgleich & Übernahme
6 Verteilte Discovery: Collector an entfernten Standorten
6.1 Einen Collector anlegen
6.2 Den Collector am Standort installieren
6.3 Was der Collector meldet
6.4 Betrieb und Überwachung
7 Client-Inventar: Funde prüfen und übernehmen
7.1 Übernehmen
7.2 Verwerfen
8 AD/LDAP – immer read-only
9 Action1-Anbindung (optional)
9.1 Was Action1 liefert — und was nicht
9.2 Einrichten
9.3 ⛔ Zwei harte Grenzen
9.4 Vom Befund zum Vollzug
10 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- CMDB aktuell und vollständig halten, ohne alles manuell zu pflegen.
- Grundlage für Lizenz-Compliance (installierte Software).
- Erkennung neuer/unbekannter Geräte im Netz.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2 Scan-Arten

Scan	Liest	Protokoll
Netzwerk-Scan	erreichbare Adressen	ICMP/Port
Deep-Scan Windows	OS, Hardware, installierte Software (Registry-Uninstall)	WMI / PowerShell
Deep-Scan Linux/Unix	System- und Paketinfos (dpkg/rpm)	SSH
Deep-Scan Netzwerkgerät	Systembeschreibung/-name	SNMP (v2c)
Deep-Scan Verzeichnis	Computerobjekte aus dem AD (read-only)	LDAP/AD

3 Anmeldedaten (Credential-Vault)

Der Deep-Scan benötigt je Ziel passende Zugangsdaten:

- **Windows (WMI/PowerShell):** `DOMAIN\Benutzer` + Passwort.
- **SSH:** Benutzer + Passwort **oder** privater Schlüssel.
- **SNMP:** Community-String (v2c).
- **LDAP/AD:** Bind-DN bzw. Domänen-Benutzer + Passwort.

Zugangsdaten werden **AES-256-verschlüsselt** im **Credential-Vault** gespeichert (Klartext wird nie zurückgegeben).

So hinterlegen Sie Zugangsdaten:

> 💻 **Nur im Client:** *CMDB* ▸ *Discovery-Scan* ▸ *Anmeldedaten* · 🔑 `discovery.manage` — Im Portal gibt es den Vault nicht; er verlangt bewusst den verwalteten Arbeitsplatz.

🔒 **Pflichtreihenfolge** — ohne gesetzten Vault-Schlüssel (siehe die Karte unten) lässt sich nichts speichern. Das ist die erste Voraussetzung, nicht die letzte.

1. Öffnen Sie *CMDB* ▸ *Discovery-Scan* und wechseln Sie auf *Anmeldedaten*.
2. Legen Sie einen neuen Eintrag an und wählen Sie das Verfahren (Windows, SSH, SNMP, LDAP/AD).
3. Tragen Sie Benutzer und Kennwort bzw. den privaten Schlüssel oder den Community-String ein.
4. Vergeben Sie einen sprechenden Namen – er ist das, was Sie beim Scan auswählen.
5. Speichern Sie.

✅ **Ergebnis:** Der Zugang steht beim Scan zur Auswahl. Das Kennwort selbst bekommen Sie nie wieder zu sehen – auch nicht als Administrator.

> **Voraussetzung:** Der Vault-Schlüssel `CredentialVault:EncryptionKey` (32-Byte, Base64) muss als Umgebungs-/Secret-Variable gesetzt sein – **nicht** in `appsettings.json`.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Wichtig:** Die Verwaltung der Anmeldedaten erfordert das Recht `discovery.manage`.

Für **Collector an entfernten Standorten** gilt eine andere Regel: Deren Scan-Kennwort steht **nicht** in diesem Vault, sondern bleibt auf dem Standort-Rechner (Abschnitt 6.2).

4 Einen Scan durchführen

> 💻 **Client:** *CMDB* ▸ *Discovery-Scan* · 🌐 **Portal:** *Netz (IPAM)* ▸ *Discovery* · 🔑 `discovery.read` zum Lesen, `discovery.manage` zum Starten

> **Voraussetzung:** Die Anmeldedaten für das Zielsystem liegen im Credential-Vault (Abschnitt 3). Ohne sie liefert ein Scan nur, was ohne Anmeldung sichtbar ist – also fast nichts.

🔒 **Pflichtreihenfolge** — Ziel und Scan-Art bestimmen, **welche** Anmeldedaten überhaupt zur Auswahl stehen. Ein zuerst gewählter Zugang passt danach womöglich nicht zum Verfahren.

1. Öffnen Sie *CMDB* ▸ *Discovery-Scan*.
2. Wählen Sie das *Ziel* (Subnetz, Hostliste oder AD) und die *Scan-Art*.
3. Wählen Sie die passenden **Anmeldedaten** aus dem Vault.
4. Starten Sie den Scan. Die **API** führt ihn im Hintergrund aus; in einem entfernten Netz übernimmt das der dort installierte **Discovery-Collector** (Abschnitt 6).
5. Warten Sie das Ende des Laufs ab.

✅ **Ergebnis:** Die Treffer stehen im **Scan-Abgleich** als Kandidaten – noch nicht als CI. Erst das Übernehmen legt an (Abschnitt 5).

5 Scan-Abgleich & Übernahme

Im **Scan-Abgleich** vergleichen Sie Treffer mit vorhandenen CIs:

- **Neu:** legt ein neues CI an.
- **Aktualisieren:** ergänzt/aktualisiert Attribute und installierte Software.
- **Ignorieren:** blendet den Kandidaten aus.

Erst beim **Übernehmen** entstehen CIs, Software-Installationen und ein **Inventar-Snapshot**. Abbildung 1 zeigt den Weg vom Scan bis dorthin.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

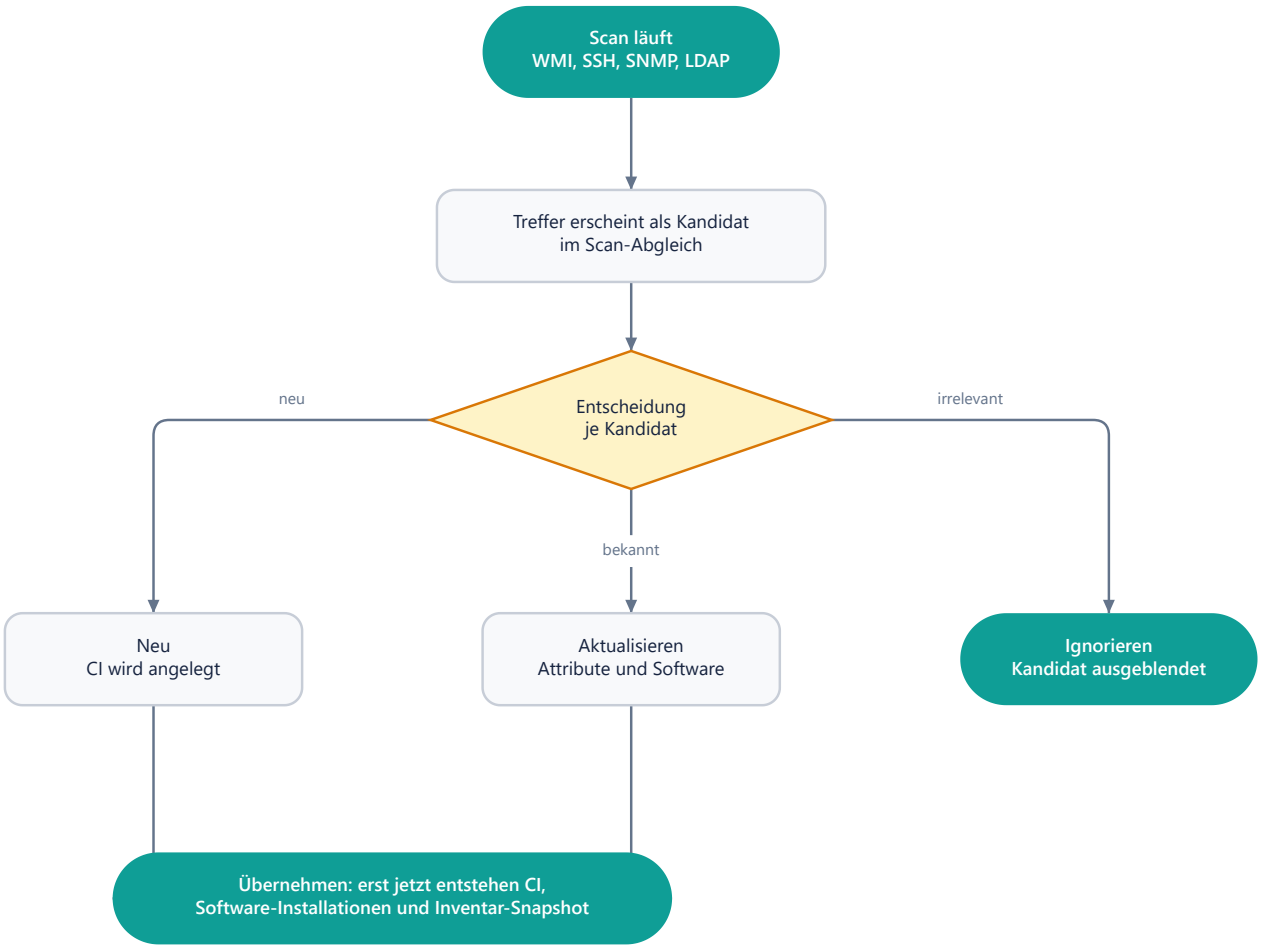

Abb. 1 – Vom Scan-Treffer zum Configuration Item. Bis zum Übernehmen ändert sich in der CMDB nichts.

6 Verteilte Discovery: Collector an entfernten Standorten

Die Scans aus Abschnitt 4 laufen **auf dem Ordivis-Server**. Damit endet ihre Reichweite an der ersten Netzgrenze: Eine Außenstelle, ein isoliertes Produktionssegment oder das Netz eines betreuten Kunden ist von dort aus nicht erreichbar – und was man nicht erreicht, kann man nicht inventarisieren.

Ein **Collector** ist ein kleiner Windows-Dienst, den Sie *im* fremden Netz installieren. Er durchsucht die Netzbereiche, die Sie ihm zuweisen, und meldet seine Funde an die Zentrale.

Die Verbindung geht immer vom Standort AUS. Der Collector ruft in der Zentrale an, nie umgekehrt. Am Remote-Standort ist deshalb **keine eingehende Firewall-Freigabe** nötig, und ein Standort hinter einem NAT-Router funktioniert genauso wie einer mit fester Adresse.

6.1 Einen Collector anlegen

💻 **Nur im Client:** *CMDB* ▸ *Discovery-Collector* ▸ *Collector anlegen* · 🔑 `discovery.manage` — Im Portal gibt es die Collector-Verwaltung nicht.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

🔒 **Pflichtreihenfolge** — der Anmeldeschein entsteht **beim Speichern** und wird **einmal** gezeigt. Wer ihn nicht sofort kopiert, muss den Collector neu anlegen.

1. Öffnen Sie *CMDB ▸ Discovery-Collector* und wählen Sie *Collector anlegen*.
2. Tragen Sie ein: - **Bezeichnung** – wie der Standort im Alltag genannt wird (z. B. „Collector Rathaus"). - **Standort** – rein beschreibend, für die Übersicht. - **Netzbereiche** – ein oder mehrere CIDR-Netze, kommagetrennt (z. B. `10.1.0.0/24, 10.1.4.0/24`). - **Abstand der Läufe** in Minuten. `0` bedeutet: läuft nur auf Anforderung, nicht von selbst.
3. Nach dem Speichern erscheint der **Anmeldeschein** – ein einmalig einlösbarer Zugangscode. **Kopieren Sie ihn jetzt:** Er wird kein zweites Mal angezeigt.

6.2 Den Collector am Standort installieren

Verwenden Sie das eigenständige Paket **Ordivis-Collector-Setup**. Es enthält ausschließlich den Collector-Dienst – nicht Server, Datenbank oder Client. Der Installer fragt:

- **Adresse der Ordivis-API** – dieselbe URL, die auch die Clients verwenden.
- **Anmeldeschein** – der Code aus Schritt 6.1.
- **Scan-Konto** (optional) – ein **lesendes** Windows-Konto in der Form `DOMÄNE\benutzer`.

Das Scan-Kennwort wird auf dem Standort-Rechner mit dem Windows-Maschinenschlüssel verschlüsselt und **verlässt ihn nie**. Die Zentrale erfährt es nicht – sie könnte es sonst über alle Standorte hinweg verlieren. Jeder Standort haftet für sein eigenes Netz.

> **Ohne Scan-Konto** bleibt es beim Netz-Durchlauf: Der Collector meldet, welche Adressen antworten, mit Name und MAC-Adresse. Das ist deutlich weniger als ein Tiefen-Scan, aber immer noch mehr als nichts.

6.3 Was der Collector meldet

Stufe	Erhebt	Voraussetzung
Netz-Durchlauf	antwortende Adressen, Hostname, MAC-Adresse, Antwortzeit	keine
Tiefen-Scan	Betriebssystem, Hersteller, Modell, Arbeitsspeicher, installierte Anwendungen	Scan-Konto (WMI)

Die **MAC-Adresse** ist dabei wichtiger, als sie aussieht: Sie ist das einzige Merkmal, das einen Rechner über einen Adresswechsel per DHCP hinweg wiedererkennt. Deshalb steht der Collector *im* Segment – von der Zentrale aus wäre sie nicht zu bekommen.

6.4 Betrieb und Überwachung

Die Liste **Discovery-Collector** zeigt je Standort den Zustand:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Zustand	Bedeutung
Anmeldung ausstehend	angelegt, aber der Anmeldeschein ist noch nicht eingelöst
Aktiv	angemeldet, meldet sich regelmäßig
Aktiv (Läufe aus)	meldet Lebenszeichen, scannt aber nicht (in der Zentrale abgeschaltet)
Überfällig	hat sich zu lange nicht gemeldet – der Rechner ist aus, der Dienst gestoppt oder die Verbindung gestört
Gesperrt	vom Betreiber gesperrt; sein Zugang ist widerrufen

Ein Collector gilt als **überfällig**, wenn seit dem letzten Lebenszeichen mehr als das Doppelte seines Melde-Abstands verstrichen ist (mindestens eine Stunde). Ein einzelnes ausgefallenes Lebenszeichen ist noch kein Vorfall.

Sperren widerruft den Zugang sofort – der Collector ist ab dem nächsten Anruf draußen, unabhängig davon, ob der Rechner noch läuft. **Anmeldeschein erneuern** setzt einen neuen Code aus und widerruft dabei den alten Zugang; das ist der Weg, um ein Geheimnis planmäßig zu wechseln.

> **Verbindungsabriss ist kein Datenverlust.** Ist die Zentrale nicht erreichbar, legt der Collector seinen Bericht in eine lokale Warteschlange und reicht ihn beim nächsten Kontakt nach. Weil jeder Bericht eine eigene Kennung trägt, erkennt die Zentrale den Nachzügler und verbucht ihn nicht doppelt.

7 Client-Inventar: Funde prüfen und übernehmen

Alles, was Collector und angebundene Werkzeuge melden, landet unter **CMDB ▸ Client-Inventar** – und zwar als **Fund**, nicht als CI.

> **Ein Fund ist kein Configuration Item.** Was im Netz gefunden wurde, ist eine Beobachtung. Ein CI ist eine Aussage der Organisation über ihren Bestand. Erst die Bestätigung durch einen Menschen macht aus dem einen das andere – genau wie bei den unbestätigten IP-Adressen der IPAM.

Die Liste ist filter-, sortier- und exportierbar (CSV und PDF), mit zwei Sichten:

- **Clients** – eine Zeile je Rechner, mit Standort, Betriebssystem und Anzahl der Anwendungen.
- **Anwendungen** – eine Zeile je Anwendung und Version, mit der Zahl der betroffenen Clients. Das ist die Frage, die im Betrieb tatsächlich gestellt wird: „Wo läuft das überall?"

7.1 Übernehmen

> 💻 **Nur im Client:** *CMDB ▸ Client-Inventar* · 🔑 `cmdb.ci.update` — Im Portal gibt es das Client-Inventar nicht.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

🔒 **Pflichtreihenfolge** — der Dublettenabgleich (Schritt 3) läuft **vor** der Entscheidung. Wer ihn übergeht und blind anlegt, hat denselben Rechner zweimal in der CMDB.

1. Öffnen Sie *CMDB ▸ Client-Inventar*.
2. Markieren Sie den Fund, den Sie übernehmen möchten.
3. Klicken Sie auf *Übernehmen*. Ordivis Platform prüft **zuerst**, ob es schon ein CI mit denselben Merkmalen gibt (gleiche MAC-Adresse, gleicher Hostname), und zeigt die Treffer an.
4. Entscheiden Sie: - **Bestehendes CI** – der Fund aktualisiert dessen Merkmale und setzt den Verifizierungsstempel. - **Neues CI anlegen** – als „Client / Notebook“; die Klasse können Sie danach jederzeit ändern.
5. Bestätigen Sie.

✅ **Ergebnis:** Aus der Beobachtung ist eine Aussage der Organisation geworden – ein CI mit Verifizierungsstempel.

In beiden Fällen werden die gemeldeten Anwendungen als **Software-Installationen** mit dem Client verknüpft – die Grundlage der Lizenzzählung.

Der Abgleich steht **vor** der Übernahme und nicht als Bereinigung danach: Wer ihn übergeht, legt Dubletten an und merkt es erst, wenn beide gepflegt werden müssen.

7.2 Verwerfen

Verwerfen nimmt einen Fund vom Tisch, ohne ein CI anzulegen – für Geräte, die dokumentiert nicht in die CMDB gehören (Gastgeräte, Fremdtechnik, Testaufbauten). Die Entscheidung hält: Beim nächsten Lauf kommt derselbe Fund nicht wieder hoch. Über **Wieder öffnen** holen Sie ihn zurück.

8 AD/LDAP – immer read-only

Der Verzeichnis-Scan liest ausschließlich. Ordivis Platform schreibt **niemals** ins Active Directory.

Ebenso gilt: **LAPS-Kennwörter und BitLocker-Wiederherstellungsschlüssel werden nie übertragen.** Sie stehen im Verzeichnis teils an denselben Objekten wie die Computerdaten; in einem Betriebsdatenbestand haben sie nichts zu suchen.

9 Action1-Anbindung (optional)

Ordivis kann **Action1** als zusätzliche Erfassungsquelle nutzen — behandelt wie die Ordivis Toolbox: Die Funde landen im **selben Staging** (§7) und werden dort geprüft und übernommen. Es entsteht kein zweiter Weg in die CMDB.

⚠️ **Durchgehend optional.** Ohne eingerichtete Verbindung tut die Anbindung nichts; ein Haus ohne Action1-Konto merkt von ihr nichts.

9.1 Was Action1 liefert — und was nicht

⚠️ **Action1 speichert kein Inventar.** Die Cloud fragt den Endpunkt bei Bedarf ab, hält das Ergebnis kurz vor und löscht es danach wieder; dauerhaft geführt werden nur Rechnername, IP-Adresse und

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Verbindungsstatistik. Es gibt deshalb kein „Inventar lesen", sondern nur *Abfrage auslösen → warten → Ergebnis abholen*. Ordivis bildet das als **Auftrag** ab und führt den Bestand anschließend selbst weiter: **Action1 liefert den Augenblick, Ordivis die Geschichte.**

⚠️ **Jede Abfrage geht auf die Endgeräte los** und ist damit nicht umsonst. Der Abrufabstand ist deshalb eine Betriebsentscheidung und steht in der Verbindung (Untergrenze 15 Minuten).

⚠️ **Action1 erreicht nur Geräte mit eigenem Agenten.** Netzwerktechnik, Appliances und alles ohne Agent bleiben außen vor. Die **Abdeckungsanzeige** nennt das ausdrücklich — eine unsichtbare Lücke ist im Audit schlimmer als eine bekannte.

⚠️ **Schwachstellen löst Action1 gegen NIST NVD, MSRC und VulnCheck auf, nicht gegen CERT-Bund.** Für BSI-Nachweise bleibt der CSAF-Weg notwendig; die Anbindung ersetzt ihn nicht.

9.2 Einrichten

Einstellungen › Erkennung › Action1 — erforderlich ist das Recht `discovery.action1.admin`.

Angabe	Bedeutung
Region	NorthAmerica, NA-2, **Europe** (Vorgabe für den DACH-Raum), Australia. Sie bestimmt das Rechenzentrum.
Client-ID / Client-Secret	OAuth2-Zugangsdaten. Das Geheimnis wird verschlüsselt abgelegt und nie im Klartext zurückgegeben.
Organisation	Action1 führt eine eigene Mandantenebene. Die Zuordnung wird **ausgewählt, nie geraten** — eine falsche trüge Befunde eines fremden Kunden in Ihren Bestand.
Abrufabstand	Wie oft abgeglichen wird (Vorgabe 12 Stunden).
Anfragen je Minute	Schutz vor der Ratengrenze des Anbieters.

Die Verbindungsprüfung meldet sich an und ruft `GET /Me` — sie belegt die Erreichbarkeit, nicht die Vollständigkeit der Rechte.

9.3 ⛔ Zwei harte Grenzen

Der BitLocker-Wiederherstellungsschlüssel wird gar nicht erst abgefragt. Am CI steht ausschließlich der *Zustand*, dass eine Verschlüsselung aktiv ist. Der Schlüssel verbleibt allein im Active Directory und wird von Ordivis **niemals ausgelesen, dokumentiert oder genutzt**. Das ist strenger als „nicht speichern" und bestimmt die Bauart: Beim Abrufen und Herausfiltern liefe der Schlüssel durch Arbeitsspeicher, Zwischenspeicher und möglicherweise durch Protokolle. Die Sperre greift bei der Auswahl der Datenquellen und ist **nicht abschaltbar** — eine abschaltbare Sperre wäre keine.

Personenbeziehbare Datenquellen sind ausgeschaltet, bis jemand sie einschaltet. Angemeldete Benutzer, Anmeldestatistik, Profile, laufende Prozesse, Webbrowser, Sofortnachrichtendienste und Cloud-Speicher fallen unter die Mitbestimmung (§ 87 BetrVG bzw. die Personalvertretungsgesetze der Länder). Jede Quelle nennt ihren **Zweck**, und jedes Ein- und Ausschalten wird mit Person und Zeitpunkt protokolliert. Bei einem Behördenkunden ist das ein Projektrisiko, kein Nebenaspekt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

9.4 Vom Befund zum Vollzug

Die Rückrichtung ist bewusst schmal: Action1 ist **keine Senke für CI-Daten**. Möglich sind

- **Ausrollauftrag** aus einem Änderungsantrag — ⚠️ **erst nach Freigabe, nie automatisch**. Der Weg vom Befund zur Änderung an hunderten Arbeitsplätzen darf keine Abkürzung haben.
- **Rückmeldung je Gerät** an denselben Änderungsantrag, damit niemand in zwei Systemen nachsieht.
- **Risikoakzeptanz**: Wird ein Befund im ISMS mit Begründung akzeptiert, steht dieselbe Entscheidung auch in Action1 — eine Entscheidung, zwei Systeme konsistent. Ohne Begründung wird nichts geschrieben.
- **Wartungsfenster**: Der im Antrag geplante Termin wird übertragen, statt ihn zweimal einzutragen.

10 Rollen & Rechte (Auszug)

Recht	Erlaubt
`discovery.read`	Scan-Daten, Collector und Client-Inventar lesen
`discovery.manage`	Scans auslösen, Anmeldedaten und Anhänge verwalten
`discovery.collector.manage`	Collector anlegen, Anmeldeschein ausstellen, sperren, löschen
`discovery.result.review`	Funde prüfen, als CI übernehmen oder verwerfen
`cmdb.ci.create` / `cmdb.ci.update`	die bei der Übernahme entstehenden CIs schreiben

Die beiden Collector-Rechte sind bewusst von `discovery.manage` getrennt: Wer einen Scan auslösen darf, darf damit noch lange nicht einen Dienst in ein fremdes Netz stellen oder CIs anlegen.

Verwandte Themen

- **ID-21 – Modul CMDB** – Übernahme der Fundstücke in die CMDB
- **ID-22 – Modul IPAM** – gefundene Adressen in der IP-Verwaltung
- **ID-21b – Benutzerkonten in der CMDB** – Konten aus AD und LDAP
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Netzzugang und Dienstkonten für Scans

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · WISSENSDATENBANK

Modul Wissensdatenbank

Wissen erfassen, freigeben und wiederverwenden – Selbsthilfe und KEDB

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-25 Wissen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Wissensdatenbank

In jedem IT-Team gibt es Fragen, die immer wiederkehren: Wie richte ich das VPN ein? Warum druckt der Drucker im zweiten Stock nicht? Oft kennt eine Kollegin die Antwort längst, nur steht sie nirgends, also wird sie jedes Mal aufs Neue erklärt. Das kostet Zeit und hängt am Zufall, wer gerade greifbar ist.

Eine Wissensdatenbank (englisch *Knowledge Base*, kurz KB) ist eine geordnete Sammlung solcher Antworten in Form durchsuchbarer Artikel. Einmal aufgeschrieben, hilft eine Lösung beliebig oft weiter. In Ordivis Platform dient sie zwei Gruppen: Endbenutzer finden im Portal öffentliche Artikel und lösen einfache Fälle selbst; Agenten nutzen interne Artikel als Nachschlagewerk und bekommen passende Vorschläge direkt am Ticket.

INHALT

1 Zweck & Nutzen
2 Einen Artikel schreiben
3 Aus einer Ticket-Lösung einen Artikel machen
4 Freigabe-Workflow & Versionierung
4.1 Einen Artikel einreichen und freigeben
4.2 Fällige Überprüfung
5 Suchen und Vorschläge
6 Feedback, Kennzahlen und Abonnements
7 Artikel aus- und einlesen
8 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Wiederkehrende Fragen einmal beantworten, vielfach nutzen.
- Endbenutzer lösen einfache Fälle selbst (Entlastung des Service Desk).
- Bekannte Fehler (Known Errors) samt Workaround bereitstellen.

2 Einen Artikel schreiben

🖥️ **Client:** *Service Desk* ▸ *Wissensdatenbank* ▸ *Neuer Artikel* · 🌐 **Portal:** *Wissen & Hilfe* ▸ *Wissensdatenbank-Redaktion* ▸ *Neuer Artikel* · 🔑 `knowledge.author`

Ein Artikel ist nach dem Muster **Symptom** → **Lösung** aufgebaut. Das ist Absicht: Wer sucht, sucht mit dem Symptom, nicht mit der Lösung.

🔒 **Pflichtreihenfolge** — die *Sichtbarkeit* entscheidet, wer den Artikel je zu sehen bekommt. Setzen Sie sie, **bevor** Sie den Artikel einreichen; nach der Veröffentlichung ist ein öffentlicher Artikel bereits im Portal gewesen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Klicken Sie auf *Neuer Artikel*.
2. Tragen Sie unter *Titel* ein, wonach jemand suchen würde – „VPN verbindet nicht“ statt „Netzwerkkonfiguration“.
3. Beschreiben Sie unter *Symptom*, was der Betroffene sieht: Meldung, Verhalten, Umstände.
4. Beschreiben Sie unter *Lösung* den Weg – in nummerierten Schritten, nicht als Fließtext.
5. Wählen Sie eine *Kategorie*.
6. Tragen Sie unter *Schlagwörter* die Begriffe ein, unter denen gesucht wird, **kommagetrennt**. Denken Sie an das Wort des Anwenders, nicht an das der IT („WLAN geht nicht“, nicht „802.1X“).
7. Setzen Sie die *Sichtbarkeit* auf *Intern* (nur Bearbeiter) oder *Öffentlich* (auch Endbenutzer im Portal).
8. Klicken Sie auf *Speichern*.

✅ **Ergebnis:** Es erscheint *Artikel gespeichert*. Der Artikel steht als **Entwurf** in der Liste und ist noch für niemanden sichtbar.

> **Tipp:** Schreiben Sie Artikel für Endbenutzer in einfacher Sprache mit klaren Schritten – wie im Benutzerhandbuch (ID-10). Ein interner Artikel darf Fachbegriffe verwenden; ein öffentlicher nicht.

3 Aus einer Ticket-Lösung einen Artikel machen

> 💻 **Nur im Client:** Ticket-Detail ▸ *Artikel aus Lösung erstellen* · 🔑 `knowledge.author` — Im Portal legen Sie den Artikel stattdessen in der Redaktion an (Abschnitt 2).

💡 **Empfohlene Reihenfolge** — schreiben Sie den Artikel **vor** dem Schließen des Tickets. Danach fehlen erfahrungsgemäß genau die Einzelheiten, die ihn brauchbar machen.

1. Öffnen Sie das gelöste Ticket.
2. Klicken Sie auf *Artikel aus Lösung erstellen*. Inhalt und Zusammenhang werden übernommen.
3. Kürzen Sie den übernommenen Text auf das, was ein Unbeteiligter braucht.
4. Ergänzen Sie *Titel*, *Kategorie*, *Schlagwörter* und *Sichtbarkeit*.
5. Speichern Sie.

✅ **Ergebnis:** Der Artikel steht als Entwurf in der Wissensdatenbank und ist mit dem Ticket verknüpft.

4 Freigabe-Workflow & Versionierung

Artikel durchlaufen **Entwurf** → **Review** → **Veröffentlicht**. Änderungen werden **versioniert**; frühere Fassungen bleiben nachvollziehbar. Veraltete Artikel lassen sich zurückziehen. Abbildung 1 zeigt die Zustände und die Wege dazwischen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

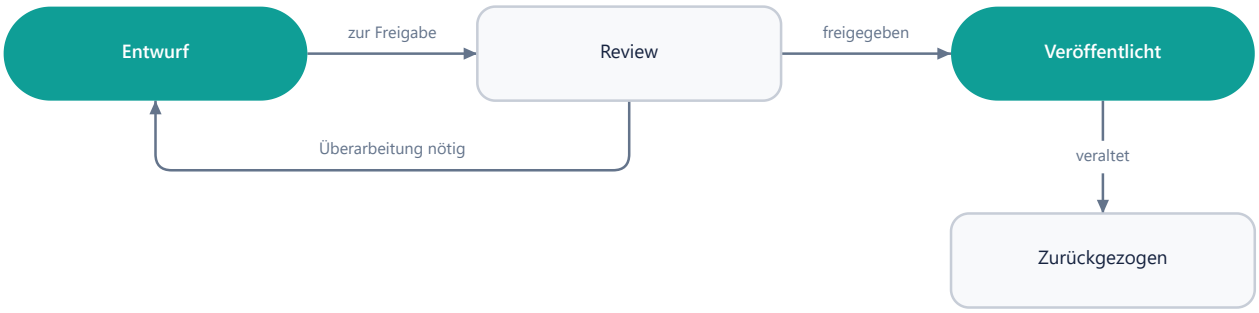

Abb. 1 – Freigabe eines Wissensartikels. Aus dem Review geht es entweder weiter zur Veröffentlichung oder zurück in den Entwurf.

4.1 Einen Artikel einreichen und freigeben

> 💻 **Client:** *Service Desk* ▸ *Wissensdatenbank*, Spalte *Aktionen* · 🌐 **Portal:** *Wissen & Hilfe* ▸ *Wissensdatenbank-Redaktion* · 🔑 `knowledge.author` zum Einreichen, `knowledge.review` zum Prüfen, `knowledge.publish` zum Veröffentlichen

🔒 **Pflichtreihenfolge** — die drei Schritte gehören drei verschiedenen Rechten. Wer nur `knowledge.author` hat, kommt bis Schritt 1 und nicht weiter; das ist das Vier-Augen-Prinzip und kein Fehler.

1. **Als Verfasser:** Klicken Sie beim Entwurf auf *Zur Prüfung einreichen*.
2. **Als Prüfer:** Öffnen Sie den Artikel, lesen Sie ihn und klicken Sie auf *Freigeben* – oder auf *Ablehnen*, dann geht er mit Begründung zurück in den Entwurf.
3. **Als Veröffentlicher:** Klicken Sie auf *Veröffentlichen*.

✅ **Ergebnis:** Der Status wechselt auf *Veröffentlicht*. Ein öffentlicher Artikel ist ab jetzt für Endbenutzer im Portal sichtbar; im Portal erscheint *Artikel veröffentlicht*.

Abkürzung für interne Artikel: Über *Direkt veröffentlichen (intern)* überspringen Sie die Prüfung. Das ist bewusst nur für **interne** Artikel vorgesehen.

Einen Artikel zurückziehen: Klicken Sie auf *Zurückziehen*. Der Artikel verschwindet aus der Suche, bleibt aber mit seiner Historie erhalten – im Gegensatz zu *Löschen*.

> **Achtung:** *Löschen* entfernt den Artikel samt Versionen endgültig. Bei einem veralteten Artikel ist fast immer *Zurückziehen* das Richtige: Es beantwortet später die Frage, warum eine Anleitung nicht mehr gilt.

4.2 Fällige Überprüfung

> 💻 **Client:** *Service Desk* ▸ *Wissensdatenbank* · 🌐 **Portal:** *Wissen & Hilfe* ▸ *Wissensdatenbank-Redaktion* · 🔑 `knowledge.author`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

In der Redaktionsliste zeigt die Spalte *Überprüfung fällig* mit *Ja*, welche Artikel turnusmäßig nachzusehen sind. Ein Artikel über eine Software von vorgestern ist schlimmer als kein Artikel – er kostet den Leser Zeit und Vertrauen.

1. Öffnen Sie die Redaktionsliste.
2. Sortieren Sie über die Spalte *Überprüfung fällig*.
3. Öffnen Sie jeden Artikel mit *Ja*, prüfen Sie ihn und speichern Sie ihn erneut.

✅ **Ergebnis:** Mit dem Speichern beginnt die Frist von vorn.

5 Suchen und Vorschläge

💻 **Client:** *Service Desk* ▸ *Wissensdatenbank* · 🌐 **Portal:** *Wissen & Hilfe* ▸ *Wissensdatenbank* · 🔑 `knowledge.read.public` bzw. `knowledge.read.internal`

1. Tippen Sie einen Begriff in das Suchfeld (im Client beschriftet *Suche (Titel, Stichwörter, Symptom)…*).
2. Grenzen Sie über *Kategorien* und *Schlagworte* weiter ein.
3. Öffnen Sie einen Treffer mit *Ansehen*.

✅ **Ergebnis:** Sie sehen den Artikel mit Symptom und Lösung.

Vorschläge ohne Suche. An zwei Stellen kommt die Wissensdatenbank von selbst:

- Am **Ticket** bekommt der Agent passende Artikel angeboten.
- Beim **Schreiben eines neuen Tickets** im Portal blendet das Formular unter der Beschreibung passende Artikel ein; über *Hat geholfen – kein Ticket nötig* endet der Vorgang, bevor er entsteht (ID-10, Abschnitt 4).

6 Feedback, Kennzahlen und Abonnements

💻 **Nur im Client:** *Service Desk* ▸ *Wissensdatenbank* ▸ *Analytics* bzw. *Abos* · 🔑 `knowledge.read.internal`

Leser geben an, ob ein Artikel geholfen hat. Daraus entstehen die Spalten *Aufrufe* und *Hilfreich* – und damit die Antwort auf die Frage, welcher Artikel sich lohnt.

1. Klicken Sie auf *Analytics*, um Aufrufe und Nutzen über alle Artikel zu sehen.
2. Nehmen Sie sich die Artikel mit vielen Aufrufen und wenig *Hilfreich* zuerst vor – dort ist der Ertrag am größten.
3. Über *Abos* (*Meine Abonnements*) lassen Sie sich benachrichtigen, wenn sich ein für Sie wichtiger Artikel ändert.

✅ **Ergebnis:** Sie überarbeiten nicht nach Gefühl, sondern nach Nutzung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

7 Artikel aus- und einlesen

> 💻 **Nur im Client:** *Service Desk* ▸ *Wissensdatenbank* ▸ *Export* bzw. *Import* · 🔑 `knowledge.admin` — Das Portal bietet keinen Massen-Export.

1. Klicken Sie auf *Export*, um **alle Artikel als ZIP** zu sichern.
2. Klicken Sie auf *Import*, um Artikel aus `.md`- oder `.zip`-Dateien einzulesen.

✅ **Ergebnis:** Die Wissensdatenbank lässt sich zwischen Mandanten übernehmen und außerhalb des Systems sichern.

> **Wichtig:** Der Export ist **kein** Ersatz für die Datensicherung des Servers (ID-35). Er enthält die Artikel, nicht die Aufrufzahlen, Freigabestände und Verknüpfungen.

8 Rollen & Rechte (Auszug)

Recht	Erlaubt
`knowledge.read.public` / `knowledge.read.internal`	Öffentliche bzw. interne Artikel lesen
`knowledge.author`	Artikel erstellen und bearbeiten
`knowledge.review`	Artikel prüfen (Review-Gate)
`knowledge.publish`	Artikel veröffentlichen und zurückziehen
`knowledge.admin`	Artikel löschen und verwalten

Verwandte Themen

- **ID-20 – Modul Service Desk / Ticketing** – Wissen am Ticket verwenden
- **ID-10 – Benutzerhandbuch – Self-Service-Portal** – was Endbenutzer im Portal davon sehen
- **ID-11 – Agentenhandbuch – Service Desk** – Artikel während der Ticketarbeit schreiben
- **ID-41 – Modul Identität, Benutzer & Rollen** – Rechte für Verfassen und Freigeben

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · WORKFLOWS

Modul Workflows & Aufgaben

Wiederkehrende Abläufe automatisieren – Aufgaben und Genehmigungen

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-26 Workflows

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Workflows & Aufgaben

Manche Abläufe wiederholen sich in fast gleicher Form. Kommt eine neue Kollegin ins Haus, sind jedes Mal dieselben Dinge zu erledigen: Konto anlegen, Notebook bestellen, Zugänge einrichten, Einweisung planen. Läuft das nur über Zuruf und Gedächtnis, wird leicht ein Schritt vergessen, oder es ist unklar, wer gerade dran ist.

Ein *Workflow* ist ein solcher Ablauf, in feste Schritte zerlegt und mit klaren Zuständigkeiten versehen. Das Modul bildet wiederkehrende Prozesse als Vorlage ab. Startet jemand den Ablauf, entstehen daraus automatisch die einzelnen Aufgaben in der richtigen Reihenfolge; erledigte Schritte lösen den nächsten aus, und wo eine Freigabe nötig ist, wartet der Ablauf auf die Entscheidung. So wird aus einem Prozess im Kopf ein nachvollziehbarer, immer gleich ablaufender Vorgang.

Wichtig – der Stand der Oberfläche. Von diesem Modul ist in Client und Portal **eine** Seite sichtbar: die eigene Aufgabenliste (Abschnitt 4). Vorlagen anlegen und bearbeiten geht derzeit **nicht über die Oberfläche**, sondern über die Schnittstelle (Abschnitt 5) beziehungsweise über die mitgelieferten Vorlagen (Abschnitt 3). Dieses Dokument beschreibt, was wirklich da ist – ausdrücklich auch, was fehlt. Für Abläufe, die Sie selbst zusammenstellen möchten, sind heute die **Checklisten** (ID-29a) der richtige Weg: Dort gibt es einen vollständigen Vorlagen-Editor in beiden Oberflächen.

INHALT

1 Zweck & Nutzen
2 Wie ein Workflow aufgebaut ist
3 Die mitgelieferten Vorlagen
4 „Meine Aufgaben“ – die eigene Aufgabe erledigen
5 Einen Workflow über die Schnittstelle starten
6 Verknüpfung mit anderen Modulen
7 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Standardisierte, nachvollziehbare Abläufe.
- Klare Zuständigkeiten über **Aufgaben**.
- Kontrollierte Freigaben über **Genehmigungen**.

2 Wie ein Workflow aufgebaut ist

Eine **Vorlage** beschreibt den Ablauf. Sie besteht aus zwei Angaben:

Wodurch sie startet (der Auslöser):

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Auslöser	Bedeutung
Manuell	Jemand startet den Ablauf bewusst
Ereignisbasiert	Ein Vorgang im System startet ihn (z. B. „Change Request angelegt“)
CI-Attribut	Ein Wert an einem CI erreicht eine Schwelle (z. B. Garantieende in 30 Tagen)

Woraus sie besteht (die Schritte, in fester Reihenfolge):

Schrittart	Was geschieht
Genehmigung	Der Ablauf hält an und wartet auf Zustimmung oder Ablehnung
Manuell	Eine Person erledigt etwas und hakt es ab
Benachrichtigung	Es geht eine Meldung heraus, der Ablauf läuft weiter
Warten	Der Ablauf pausiert eine festgelegte Zeit
Ticket anlegen	Es entsteht ein Vorgang im Service Desk
CI ändern	Ein Wert in der CMDB wird gesetzt

Je Schritt sind Zuständige (Person, Rolle oder Gruppe) und eine **Frist** hinterlegt. Läuft die Frist ab, greift die Eskalation.

3 Die mitgelieferten Vorlagen

Bei der Einrichtung eines Mandanten werden sechs Vorlagen angelegt:

Vorlage	Auslöser	Was sie tut
Change-Genehmigung (CAB)	Ereignis *Change Request angelegt*	CAB-Genehmigung (Frist 72 h) → Bestätigung → Implementierung prüfen (Frist 168 h)
Beschaffung-CI	manuell	Bedarf prüfen → Angebote einholen → Bestellung genehmigen → Wareneingang und CI-Aufnahme bestätigen
Garantie-Erinnerung	CI-Attribut	30 Tage vor Garantieablauf melden, dann Maßnahme festlegen
Vertragsablauf-Erinnerung	CI-Attribut	60 Tage vor Vertragsablauf melden
Mitarbeiter-Onboarding	manuell	Konto, Ausstattung, Zugänge, Einweisung
Mitarbeiter-Offboarding	manuell	Rückgabe, Sperrung, Übergabe

> **Hinweis:** Die Vorlagen entstehen **einmalig** bei der Einrichtung. Sie werden nicht überschrieben, wenn sie bereits vorhanden sind – eine Anpassung über die Schnittstelle bleibt also bestehen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4 „Meine Aufgaben“ – die eigene Aufgabe erledigen

💻 **Client:** *Meine Arbeit* ▸ *Meine Workflow-Aufgaben* · 🌐 **Portal:** *Meine Vorgänge* ▸ *Meine Aufgaben* · 🔑 `workflow.read`

1. Öffnen Sie die Seite. Steht dort *Keine offenen Aufgaben.*, ist nichts für Sie offen.
2. Lesen Sie den Aufgabentext und den *Status:* daneben.
3. Haben Sie die Aufgabe erledigt, klicken Sie auf *Erledigt*.
4. Handelt es sich um eine **Genehmigung**, die Sie nicht erteilen wollen, klicken Sie stattdessen auf *Ablehnen*.

✅ **Ergebnis:** Es erscheint *Aufgabe erledigt.* bzw. *Aufgabe abgelehnt.* Bei *Erledigt* löst der Ablauf den nächsten Schritt aus; bei *Ablehnen* endet er an dieser Stelle.

Abbildung 1 zeigt den Durchlauf von der Vorlage bis zum Abschluss.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

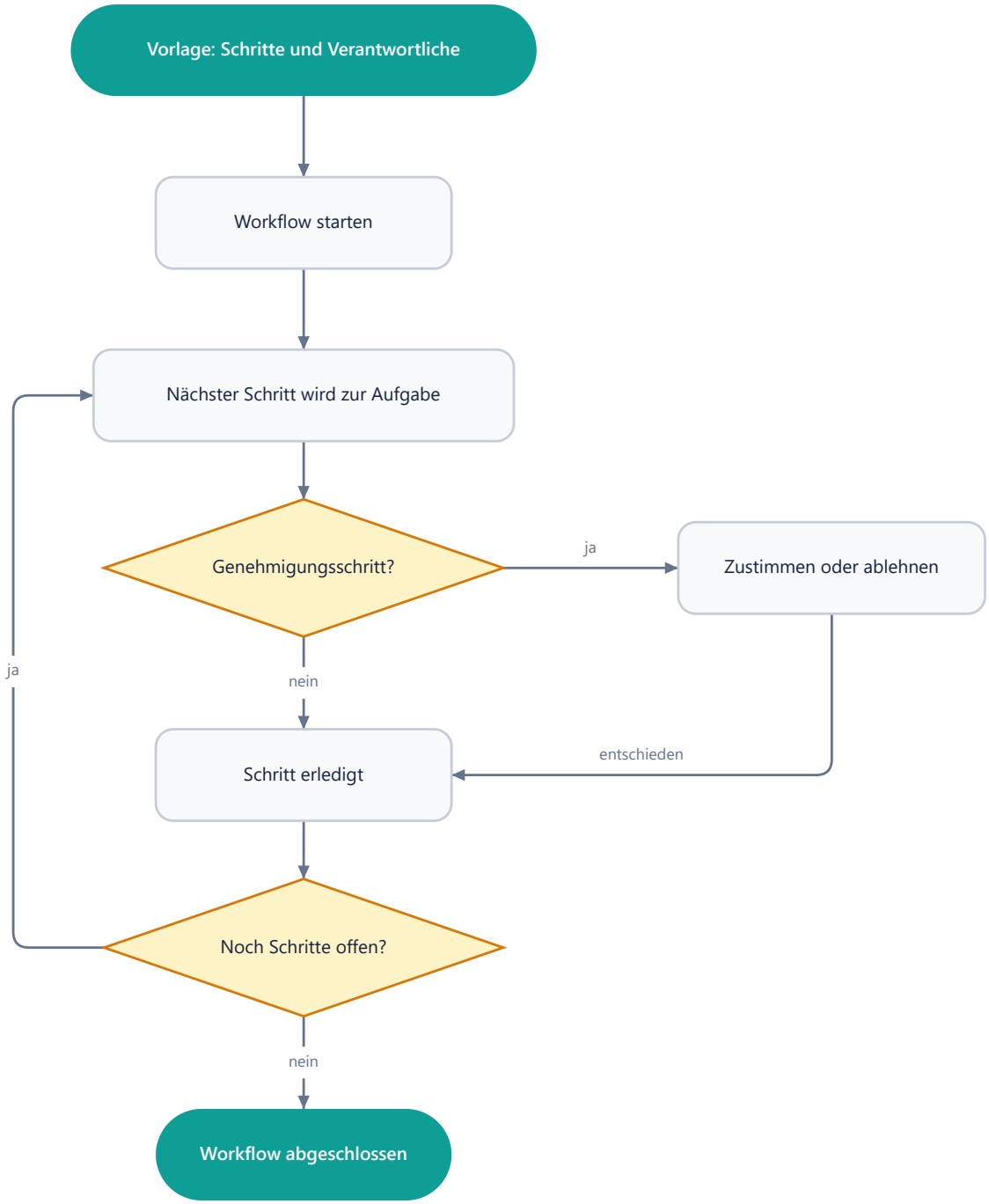

Abb. 1 – Ein Workflow von der Vorlage bis zum Abschluss. Die Schleife läuft so lange, bis kein Schritt mehr offen ist.

Achtung – zwei verschiedene Aufgabenlisten, und beide Oberflächen benennen sie anders. Die **Workflow-Aufgaben** dieses Moduls heißen im Portal *Meine Vorgänge ‣ Meine Aufgaben*, im Client *Meine Arbeit ‣ Meine Workflow-Aufgaben*. Die **selbst angelegten** Aufgaben der Selbstorganisation (ID-34) heißen umgekehrt im Portal *Meine Vorgänge ‣ Meine Aufgabenliste*, im Client *Meine Arbeit ‣ Meine Aufgaben*. Der Name *Meine Aufgaben* führt also je nach Oberfläche zu etwas anderem – in beiden steht er bei dem Eintrag, der dort zuerst da war.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Wichtig:** *Ablehnen* beendet den Ablauf, es stellt ihn nicht zurück. Wollen Sie nur eine Rückfrage stellen, tun Sie das außerhalb des Workflows – etwa über ein Ticket – und lassen Sie die Aufgabe offen.

5 Einen Workflow über die Schnittstelle starten

> 🔑 `workflow.manage` — Für diesen Weg gibt es **in keiner Oberfläche** eine Schaltfläche. Er ist für die Anbindung eigener Werkzeuge gedacht; die vollständige Beschreibung steht in ID-60.

🔒 **Pflichtreihenfolge** — Sie brauchen die Kennung der Vorlage, bevor Sie sie starten können. Ohne Schritt 1 fehlt die `templateId` in Schritt 2.

1. Rufen Sie `GET /api/v1/workflow/templates` ab und suchen Sie die gewünschte Vorlage samt ihrer Kennung heraus.
2. Rufen Sie `POST /api/v1/workflow/templates/{id}/start` auf.
3. Verfolgen Sie den Lauf über `GET /api/v1/workflow/instances/{id}` und die Schritte über `GET /api/v1/workflow/instances/{id}/tasks`.

✅ **Ergebnis:** Es entsteht ein Lauf, dessen erste Aufgabe bei der zuständigen Person in ihrer Aufgabenliste erscheint (Abschnitt 4).

> **Tipp:** Ereignis- und CI-Attribut-gesteuerte Vorlagen brauchen diesen Aufruf nicht – sie starten von selbst. Der manuelle Start ist nur für die drei Vorlagen mit Auslöser *manuell* nötig.

6 Verknüpfung mit anderen Modulen

Workflows knüpfen an Tickets, Changes und CIs an. Zwei Verbindungen sind eingerichtet:

- Ein **Change Request** startet die CAB-Genehmigung, ohne dass jemand etwas anstößt.
- Ein **CI-Attribut** (Garantie, Vertragsende) startet die passende Erinnerung.

Zu einem Ticket sehen Sie die zugehörigen Läufe über `GET /api/v1/workflow/tickets/{ticketId}/instances`.

7 Rollen & Rechte (Auszug)

Recht	Erlaubt
`workflow.read`	Workflows/Aufgaben sehen
`workflow.manage`	Workflows starten, Aufgaben bearbeiten und Vorlagen verwalten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Verwandte Themen

- **ID-29a – Modul Checklisten** – der Weg für selbst zusammengestellte Abläufe, mit Vorlagen-Editor
- **ID-20 – Modul Service Desk / Ticketing** – Changes, die die CAB-Genehmigung auslösen
- **ID-34 – Modul Selbstorganisation** – die anderen „Meine Aufgaben“
- **ID-41 – Modul Identität, Benutzer & Rollen** – Rollen für Genehmigungen
- **ID-60 – API-Referenz** – die Endpunkte aus Abschnitt 5 im Zusammenhang

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · REPORTING

Modul Reporting & Dashboards

Rollenbasierte Dashboards, Berichte, Zeitpläne und Audit-Log

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-27 Reporting

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Reporting & Dashboards

Die eigentliche Arbeit passiert in den einzelnen Modulen: Tickets werden bearbeitet, CIs gepflegt, Verträge verwaltet. Wer das Ganze steuern will, braucht aber den Überblick darüber – wie viele Tickets offen sind, ob die vereinbarten Zeiten eingehalten werden, welche Verträge bald auslaufen. Diese Zahlen aus jedem Modul einzeln zusammenzusuchen, ist mühsam und immer nur eine Momentaufnahme.

Reporting bündelt genau diese Auswertungen an einer Stelle. Ein *Dashboard* ist eine Übersichtsseite, die die wichtigsten Kennzahlen laufend und auf einen Blick zeigt, zugeschnitten auf die jeweilige Rolle. Ein *Bericht* ist eine ausführlichere Auswertung, die Sie filtern und als Datei (Excel, CSV oder PDF) exportieren können, etwa um sie in einer Besprechung zu zeigen. Berichte lassen sich auch zeitgesteuert erzeugen und automatisch per E-Mail verschicken.

INHALT

1 Rollenbasierte Dashboards
- 1.1 Wie die SLA-Erfüllung gerechnet wird
- 1.2 Was die Kachel „SLA verletzt" zählt

2 Berichte (Report-Verwaltung)
3 Geplante Berichte
4 Tabellen-Funktionen (überall)
5 Audit-Log
6 Wallboard (Anzeigegeräte)
- 6.1 Installieren
- 6.2 Einrichten
- 6.2.1 Dauerbetrieb: Was einen Schirm über Nacht abschaltet
- 6.3 Im Betrieb

7 Rollen & Rechte (Auszug)
Verwandte Themen

1 Rollenbasierte Dashboards

Jeder Bereich beginnt mit seinem Dashboard – es steht überall an erster Stelle im Menü. Der Aufbau ist in allen Bereichen derselbe: oben eine Leiste mit den wichtigsten Kennzahlen, darunter Diagramme (Verläufe und Verteilungen), darunter die **Arbeitslisten** (überfällige und neue Tickets, Prioritätsverteilung). Ein Klick auf eine Kennzahl öffnet die passende **gefilterte** Liste – „SLA verletzt" etwa genau die betroffenen Tickets, „Offen" alle offenen. Die Listen sind als kompakte Tabellen mit Spaltenüberschrift dargestellt; ein Klick auf eine Zeile öffnet den Datensatz.

Dashboard	Inhalt
Start	Eigene offene/neue Tickets, überfällige und zugewiesene Vorgänge, Aufgaben und ausstehende Changes; darunter je eine Zeile pro Bereich

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Dashboard	Inhalt
Service Desk	Kennzahlen (SLA-Erfüllung, mittlere Lösungszeit, Zufriedenheit, Durchsatz, Backlog nach Alter, Agentenlast, CSAT) **und** die Arbeitslisten: überfällige/neue Tickets, Prioritätsverteilung
CMDB	CI-Bestand, Lebenszyklus-Verteilung, zuletzt geänderte CIs, Zugänge in Staging
IPAM	Netze, vergebene Adressen, Auslastung je Netz, Adresskonflikte
Informationssicherheit	Zielobjekte, offene Maßnahmen, Schutzbedarf- und Risikoverteilung, Fortschritt des Grundschutz-Checks

Der Auswertungszeitraum lässt sich auf 7, 30 oder 90 Tage stellen.

1.1 Wie die SLA-Erfüllung gerechnet wird

Beide SLA-Kennzahlen zählen ausschließlich Vorgänge, für die es überhaupt eine Frist gibt, und sie zählen dasselbe: **Wie viele davon wurden innerhalb ihrer Frist fertig?**

- **Lösung:** Anteil der gelösten oder geschlossenen Vorgänge, deren **Lösungsfrist** nicht überschritten wurde.
- **Reaktion:** Anteil der Vorgänge mit **Reaktionsfrist** und erfolgter erster Antwort, bei denen diese Frist gehalten wurde.

Maßgeblich ist allein die Frist. Insbesondere gilt:

- Ob ein Vorgang **eskaliert** wurde, spielt für die Kennzahl keine Rolle. Eine von Hand ausgelöste Eskalation oder ein erklärter **Major Incident** sind Handlungen der Bearbeitung, keine Fristverletzung – ein so behandelter, aber pünktlich gelöster Vorgang zählt als eingehalten.
- Umgekehrt zählt eine überschrittene Frist auch dann als Verfehlung, wenn wegen der eingestellten **Wartezeiten der Eskalationsstufen** noch niemand benachrichtigt wurde.
- Verstreicht eine Frist während eines laufenden **Wartungsfensters**, gilt sie nicht als verletzt – ein geplanter Ausfall ist kein Versäumnis.

> **Hinweis zu älteren Auswertungen:** Bis Juli 2026 wurde die Lösungs-Kennzahl ersatzweise an der Eskalationsstufe gemessen und war dadurch in beide Richtungen ungenau. Mit der Umstellung wurden die vorhandenen Vorgänge und die gespeicherte Verlaufshistorie **einmalig nachgerechnet**. Zahlen aus Berichten, die vor der Umstellung erstellt wurden, können deshalb von den heutigen abweichen. Für Zeiträume vor der Umstellung ließen sich damalige Wartungsfenster nicht rekonstruieren; die Nachrechnung ist dort im Zweifel strenger und beschönigt nichts.

1.2 Was die Kachel „SLA verletzt" zählt

Die Kachel ist keine Quote, sondern eine Arbeitsmenge: **offene Vorgänge, deren Lösungsfrist der Prüflauf als gerissen festgehalten hat.** Ein Klick darauf öffnet genau diese Liste, und dieselben Vorgänge sind auch in der Ticketliste rot markiert – Kachel, Liste und Markierung meinen dasselbe.

Maßgeblich ist die Feststellung des Prüflaufs, nicht die bloße Uhrzeit. Daraus folgt:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- In den **Warte-Zuständen** („Warte auf Melder", „Warte auf Dienstleister", „Blockiert") ruht die SLA-Uhr. Dort wird keine Verletzung festgestellt, und beim Verlassen des Warte-Zustands wandert die Frist um die Wartedauer nach hinten. Eine **von Hand gesetzte Fälligkeit** wandert nicht mit – sie ist eine Zusage des Bearbeiters und soll halten, was an ihr steht.
- Eine **vor** dem Warten festgestellte Verletzung bleibt bestehen. Der Vorgang zählt weiter mit, solange er offen ist: Die Frist wurde gerissen, das Warten macht das nicht ungeschehen.
- Ein **gelöster oder geschlossener** Vorgang zählt nicht mehr mit, auch wenn seine Frist längst in der Vergangenheit liegt. War er bei der Bearbeitung verletzt, erscheint das in der SLA-Erfüllung (1.1) – die Kachel fordert zum Handeln auf und meint deshalb nur, was noch offen ist.
- Ein laufendes **Wartungsfenster** verhindert die Feststellung; ein geplanter Ausfall ist kein Versäumnis.

Der Prüflauf arbeitet im Minutentakt. Eine gerade eben abgelaufene Frist erscheint deshalb mit bis zu einer Minute Verzögerung.

Die Verlaufsdiagramme stützen sich auf tägliche Kennzahl-Schnappschüsse. Nach einer Neuinstallation entsteht diese Historie erst mit der Zeit; solange zu wenige Tage vorliegen, weist das Diagramm darauf hin, statt eine Linie aus einem einzigen Punkt zu zeichnen.

Die Auswertungskarten setzen das Recht `reporting.read` voraus. Fehlt es, blendet das Dashboard sie aus und zeigt nur die Kennzahlen, die der Benutzer ohnehin sehen darf.

Das **persönliche Dashboard** nach dem Login (Menüpunkt „Start") zeigt zusätzlich Ihre eigenen Tickets, Aufgaben, Wiedervorlagen und Genehmigungen.

2 Berichte (Report-Verwaltung)

> 💻 **Client:** *Compliance & Reporting* ▸ *Report-Verwaltung* · 🌐 **Portal:** *Compliance & Reporting* ▸ *Berichte* und *Bericht-Baukasten* · 🔑 `reporting.read` zum Lesen, `reporting.manage` zum Anlegen und Ändern

🔒 **Pflichtreihenfolge** — Spalten, Filter und Sortierung wirken auf die gewählte **Datenquelle**. Wird sie nachträglich gewechselt, ist die Zusammenstellung hinfällig.

1. Öffnen Sie die Report-Verwaltung (im Portal den *Bericht-Baukasten*).
2. Legen Sie eine **Report-Definition** an und wählen Sie die Datenquelle – optional mit eigenem SQL (ID-27a).
3. Wählen Sie **Spalten, Filter und Sortierung**.
4. Prüfen Sie das Ergebnis in der **Vorschau**, bevor Sie speichern.
5. **Exportieren** Sie als **Xlsx**, **CSV** oder **PDF**.

✅ **Ergebnis:** Der Bericht steht in der Liste und lässt sich jederzeit erneut ausführen – und zeitgesteuert versenden (Abschnitt 3).

Sie fangen nicht bei null an. Ordivis Platform bringt eine Bibliothek gebrauchsfertiger Berichte über alle Module mit – Vorgänge und Fristen, Gerätebestand, Lizenzen und Verträge, Informationssicherheit,

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Datenschutz, Organisation. Sie stehen nach der Einrichtung ohne Zutun in der Liste und lassen sich wie eigene Berichte anpassen oder als Vorlage kopieren.

> **Hinweis zu Anlagen, die vor 2026.08.02 eingerichtet wurden.** Bis zu dieser Fassung landeten diese Berichte durch einen Fehler in der falschen Datenbank und fehlten deshalb in Ihrer Liste – auch die Checklisten-Vorlagen, der Gefährdungskatalog, der ISO-27001-Katalog, der Datenschutz-Grundkatalog und die Muster-Aufbauorganisation waren davon betroffen. Mit dem Update werden sie beim nächsten Start nachgetragen. Berichte, die Sie **selbst gelöscht** haben, kommen dabei nicht zurück.

3 Geplante Berichte

Berichte lassen sich **zeitgesteuert** ausführen (Cron) und automatisch an **E-Mail-Empfänger** versenden – z. B. ein wöchentlicher SLA-Bericht an die Leitung.

4 Tabellen-Funktionen (überall)

Alle Tabellen in Ordivis Platform sind voll ausgestattet: **filtern, sortieren, Spalten anpassen, Spaltenauswahl** sowie **CSV-/PDF-Export**. Sie arbeiten also überall gleich.

5 Audit-Log

> 💻 **Client:** *Compliance & Reporting* ▸ *Audit-Log* · 🌐 **Portal:** *Compliance & Reporting* ▸ *Audit-Log* · 🔑 `audit.read`, für die Ausgabe zusätzlich `audit.export`

Das **Audit-Log** zeigt revisionssicher sicherheitsrelevante Aktionen mit Filter (Entitätstyp, Zeitraum) und Paginierung.

So finden Sie einen bestimmten Vorgang:

💡 **Empfohlene Reihenfolge** — grenzen Sie zuerst den Zeitraum ein, dann den Entitätstyp. Anders herum blättern Sie durch Tausende Einträge, bevor der Filter greift.

1. Öffnen Sie das Audit-Log.
2. Setzen Sie den **Zeitraum** möglichst eng.
3. Wählen Sie den **Entitätstyp** (z. B. Benutzer, Ticket, CI).
4. Blättern Sie durch das Ergebnis.

✅ **Ergebnis:** Sie sehen, wer wann was geändert hat – nachweisbar und ohne Lücke.

> **Wichtig:** Der Zugriff erfordert `audit.read`, die **Ausgabe als CSV** zusätzlich `audit.export`. Lesen und Herausgeben sind bewusst getrennt: Eine ausgegebene Datei verlässt das System. Einzelheiten zum Export (Filter, Vollständigkeitsangabe, Obergrenze) stehen im Administrationshandbuch.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6 Wallboard (Anzeigegeräte)

Das **Ordivis Wallboard** ist ein eigenes Vollbild-Programm für Bildschirme, die unbeaufsichtigt laufen – im Service-Desk-Büro, am Empfang, im Leitstand. Es zeigt im Wechsel große Kennzahlen und ist bewusst kein zweites Fenster des Arbeitsplatz-Programms: Es braucht keine Anmeldung durch einen Menschen, startet nach einem Stromausfall von allein wieder und lässt sich nicht bedienen wie ein Arbeitsplatz.

6.1 Installieren

Das Wallboard ist eine **abwählbare Komponente des Installers**: „Wallboard (Vollbild-Dashboards für Anzeigegeräte)". Sie ist keinem Installationstyp zugeordnet und muss bewusst angehakt werden – auf einem gewöhnlichen Arbeitsplatz wird sie also nicht mitinstalliert. Nach der Installation liegt eine Verknüpfung **Ordivis Wallboard** im Startmenü.

6.2 Einrichten

Beim ersten Start erscheint die Einrichtung (später jederzeit mit **Esc** erreichbar):

- **Serveradresse** – z. B. `https://server.firma.local:5001`. Mit **Verbindung testen** prüfen. Für interne Netze ohne eigene PKI lässt sich die Zertifikatsprüfung abschalten; verwenden Sie das nur dort.
- **Wallboard-Token** – wird vorher im Arbeitsplatz-Programm ausgestellt (siehe Betriebshandbuch, **7.4.2 Anzeigegeräte**). Er ersetzt Benutzername und Kennwort.
- **Dashboards** – Auswahl und Reihenfolge. Angehakte Dashboards laufen von oben nach unten durch: **Service Desk**, **CMDB**, **IP-Verwaltung**.
- **Takt** – wie viele Sekunden ein Dashboard stehen bleibt und wie oft die Zahlen neu geholt werden.
- **Darstellung** – dunkles Erscheinungsbild sowie, wie der Mandant oben links erscheint (Logo und Name, nur Name, nur Logo). Ist im Mandanten ein Logo hinterlegt, rückt die Ordivis-Marke klein nach unten rechts.
- **Im Vollbild öffnen auf** – der Bildschirm, auf dem das Wallboard aufgeht. An einem Anzeigegerät hängen oft zwei Schirme, einer an der Wand und einer am Arbeitsplatz; ohne diese Angabe öffnet das Wallboard dort, wo Windows das Fenster platziert. Ist der gewählte Bildschirm nicht angeschlossen, bleibt es beim aktuellen — auf einen Bildschirm zu springen, den es nicht gibt, hieße, das Fenster ins Nichts zu schieben.
- **Beim Anmelden dieses Windows-Kontos automatisch starten** – trägt das Wallboard für den angemeldeten Benutzer ein (kein Eingriff in Dienste oder Gruppenrichtlinien).
- **Nach einem unerwarteten Fehler weiterlaufen** – empfohlen für Schirme im Dauerbetrieb (Vorgabe: an). Der Fehler wird protokolliert, das Bild zeichnet sich beim nächsten Takt neu. Zum Einrichten eines neuen Anzeigegeräts kann es hilfreich sein, dies abzuschalten.

6.2.1 Dauerbetrieb: Was einen Schirm über Nacht abschaltet

Ein Wandschirm hat niemanden davor, der auf „Wiederholen" klickt. Drei Dinge greifen deshalb ineinander:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Vorkehrung	Wirkung
Weiterlaufen nach Fehler	Ein Fehler in der Oberfläche beendet den Schirm nicht mehr, sondern landet im Protokoll.
Lebenszeichen	Alle 15 Minuten hält das Protokoll fest, dass der Schirm läuft.
Geordnetes Ende	Wird der Schirm geschlossen, steht das ausdrücklich im Protokoll.

Das Lebenszeichen ist der Schlüssel zur Fehlersuche: Steht die letzte Zeile um 03:12 und fehlt der Vermerk über das geordnete Ende, wurde der Schirm um diese Zeit **abgebrochen** — und die Uhrzeit sagt meist schon, wodurch. Das Protokoll liegt unter `%ProgramData%\InfraDesk\Logs\wallboard-JJJJMMTT.log`.

⚠️ **Der häufigste Grund ist kein Fehler des Programms.** Der automatische Start hängt an der **Anmeldung** dieses Windows-Kontos — das ist nötig, weil der Token an dieses Konto gebunden ist (DPAPI, siehe Betriebshandbuch). Startet Windows nachts wegen Updates neu, steht der Rechner am Anmeldebildschirm, und der Schirm bleibt bis zur nächsten Anmeldung dunkel. Für den Dauerbetrieb empfiehlt sich eine **automatische Anmeldung** des Anzeigekontos.

6.3 Im Betrieb

Taste	Wirkung
Esc	Einstellungen öffnen
Leertaste	Wechsel anhalten / fortsetzen
Pfeiltasten	ein Dashboard vor / zurück
F5	Zahlen sofort neu holen

Wartet mindestens ein **neues Ticket**, blendet die zugehörige Kachel ruhig auf und ab – auffällig genug, um über den Raum bemerkt zu werden, ohne zu blinken.

Was der Schirm zeigt, bestimmt das Konto hinter dem Token. Ein Wallboard im Flur ist für jeden Vorbeigehenden sichtbar. Binden Sie den Token deshalb an ein eigenes Konto mit ausschließlich lesenden Rechten, nicht an ein persönliches.

7 Rollen & Rechte (Auszug)

Recht	Erlaubt
`reporting.read`	Berichte und Dashboards ansehen
`reporting.create`	Eigene Berichte erstellen
`reporting.manage`	Alle Berichte und Zeitpläne verwalten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Recht	Erlaubt
`reporting.dashboard.manage`	Dashboard-Widgets konfigurieren
`audit.read` / `audit.export`	Audit-Log ansehen / exportieren
`system.config.manage`	Wallboard-Token ausstellen und widerrufen

Verwandte Themen

- **ID-27a – Reports & SQL-Abfragen** – eigene Reports bauen und per SQL abfragen
- **ID-52 – Datenbank-Referenz** – Tabellen und Felder für eigene Abfragen
- **ID-44 – Rollen- & Rechte-Matrix** – warum zwei Benutzer verschiedene Zahlen sehen
- **ID-20 – Modul Service Desk / Ticketing** – Herkunft der Service-Desk-Kennzahlen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

ANLAGE · REPORTING

Reports & SQL-Abfragen

Reports zusammenstellen, auswerten und als Profi per SQL abfragen

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-27a Reports & SQL

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Reports & SQL-Abfragen

Der **Report-Baukasten** erzeugt Auswertungen über alle Module – vom fertigen Standard-Report für die Leitung bis zur freien SQL-Abfrage für Profis. Sie finden ihn unter **Berichte ▸ Report-Verwaltung** (Berechtigung `reporting.read`).

INHALT

1 Standard-Report-Bibliothek
2 Einen Report zusammenstellen (Baukasten)
3 Gruppieren & Kennzahlen
4 SQL-Abfragen (für Profis)
5 Kopf- und Fußzeile der Berichte
6 Berechtigungen
Verwandte Themen

1 Standard-Report-Bibliothek

Ordivis Platform liefert je Mandant **mindestens fünf fertige Reports pro Modul** aus – die von Leitung und Führungskräften regelmäßig abgefragten Auswertungen. Sie erscheinen automatisch in der Report-Verwaltung (gekennzeichnet als *System*). Beispiele:

- **CMDB:** CI-Bestand nach Klasse, aktive CIs, auslaufende Garantie/Support, *CIs je Klasse (Anzahl)*.
- **Service-Desk:** offene Tickets nach Priorität, SLA-Verletzungen, Backlog-Entwicklung, *Tickets je Status*.
- **IPAM:** IP-Bestand nach Netz, DHCP-Adressen, Netzübersicht, *IP-Adressen je Netz (Anzahl)*.
- **Assets/Verträge:** Vertragsübersicht, bald auslaufende Verträge (90 Tage), Lizenzbestand.
- **Security/ISMS:** SoA-Abdeckung, nicht erfüllte Kontrollen, Risiken nach Bewertung, offene Nichtkonformitäten.
- **BCM, Datenschutz, Identität, Checklisten, Audit:** je eigene Führungs-Reports (VVT-Übersicht, Benutzer je Rolle, überfällige Checklisten, letzte Änderungen u. v. m.).

> **Tipp:** Ein Standard-Report lässt sich als Vorlage nutzen: **Bearbeiten** öffnet ihn im Baukasten; beim Speichern entsteht eine **Kopie** als eigener Report – die Systemvorlage bleibt unverändert.

Jeder Report kann direkt **exportiert** (Excel/CSV/PDF) oder per **Zeitplan** automatisch erzeugt und per E-Mail versendet werden (Cron-Ausdruck + Empfänger, Berechtigung `reporting.manage`).

2 Einen Report zusammenstellen (Baukasten)

> 💻 **Client:** *Compliance & Reporting ▸ Report-Verwaltung ▸ Neu* · 🌐 **Portal:** *Compliance & Reporting ▸ Bericht-Baukasten* · 🔑 `reporting.read` zum Lesen, `reporting.manage` zum Speichern

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

🔒 **Pflichtreihenfolge** — die **Datenquelle** steht am Anfang. Sie bestimmt, welche Felder unter *Verfügbare Felder* überhaupt erscheinen; ein späterer Wechsel wirft die Zusammenstellung weg.

Neu öffnet den Baukasten. Vergeben Sie **Name** und **Beschreibung** und wählen Sie die **Datenquelle (Modul)** – z. B. *Configuration Items (CMDB)* oder *Tickets (Service-Desk)*. Danach:

1. **Spalten** – Ziehen Sie Felder aus **Verfügbare Felder** per Drag & Drop nach **Ausgewählte Spalten** (Doppelklick geht auch). Mit ▲▼ ordnen, mit ✕ entfernen.
2. **Filter** – Zeile hinzufügen: Feld → Operator → Wert. Die angebotenen Operatoren richten sich nach dem Datentyp (z. B. *enthält* bei Text, *in den nächsten … Tagen* / *überfällig* bei Datum, *ist* bei Auswahl-Feldern). Mehrere Filter werden mit **UND** verknüpft.
3. **Sortierung** – ein oder mehrere Felder auf- oder absteigend.
4. **Max. Zeilen** – Obergrenze für das Ergebnis.
5. **Vorschau** – zeigt die ersten 200 Zeilen sofort an, ohne etwas zu speichern.
6. **Speichern** – legt den Report an (oder aktualisiert ihn).

✅ **Ergebnis:** Der Report steht in der Liste und lässt sich ausführen, exportieren und zeitgesteuert versenden (ID-27, Abschnitt 3).

> **Wichtig:** Auswahl-Felder (z. B. Status, Priorität, Lebenszyklus) zeigen und filtern **deutsche Klartexte** statt technischer Zahlen – im Report wie im Filter.

3 Gruppieren & Kennzahlen

Der Schalter **Gruppieren** verwandelt den Report in eine **Kennzahl-Auswertung**:

- **Gruppieren nach** – ein oder mehrere Felder (z. B. *CI-Klasse*, *Status*, *Anbieter*).
- **Kennzahlen** – *Anzahl*, *Summe*, *Durchschnitt*, *Minimum*, *Maximum*. *Anzahl* zählt die Zeilen je Gruppe; *Summe*/*Durchschnitt* sind nur für Zahlenfelder möglich.

Gruppierte Reports werden automatisch nach der ersten Kennzahl absteigend sortiert – die größte Gruppe steht oben (z. B. „Verträge je Anbieter (Anzahl + Jahreskosten)“).

4 SQL-Abfragen (für Profis)

Über den Schalter **Roh-SQL (für Profis)** wechseln Sie in den freien SQL-Modus und schreiben die Abfrage direkt. Die **Vorschau** funktioniert auch hier (gedeckelt auf 200 Zeilen); Fehler werden im Klartext gemeldet.

Der Schalter erscheint nur mit der Berechtigung `reporting.create` – freies SQL läuft an den Modul-Berechtigungen vorbei und ist deshalb kein Leserecht (Begründung unter *Berechtigungen*).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Achtung: Für SQL-Abfragen gelten feste Regeln: * **Mandantenbindung:** Schränken Sie **immer** mit `WHERE … tenant_id = @tenantId` ein. Der Parameter `@tenantId` wird automatisch mit dem angemeldeten Mandanten gefüllt – ohne diese Bedingung sehen Sie Daten nicht oder mandantenübergreifend. * **Nur lesend:** Abfragen dürfen ausschließlich `SELECT` sein. * **Begrenzen:** Nutzen Sie `LIMIT`, um große Ergebnisse zu vermeiden.

Beispiel (offene Tickets je Priorität als Klartext):

```
SELECT t.priority, COUNT(*) AS anzahl
FROM ticketing.tickets t
WHERE t.tenant_id = @tenantId AND t.status NOT IN (3, 4)
GROUP BY t.priority
ORDER BY anzahl DESC
```

Tipp: Welche Tabellen und Spalten es gibt, steht in der **Datenbank-Referenz** (ID-52): Teil A listet die Baukasten-Felder je Modul mit Bedeutung, Teil B das vollständige Schema aller Tabellen und Spalten.

5 Kopf- und Fußzeile der Berichte

💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Branding* · 🔑 `system.config.manage` — Das Portal hat keine Branding-Seite; die Angaben gelten aber für **beide** Oberflächen.

Kopf- und Fußzeile der Berichtsausgabe lassen sich je Mandant anpassen:

- **Kopfzeile** – frei wählbarer Text im Kopf jedes Berichts (z. B. ein Vertraulichkeitshinweis).
- **Fußzeile** – frei wählbarer Text im Fuß jedes Berichts (z. B. Firmen-/Abteilungsangabe).

Zusätzlich tragen **alle** Berichte automatisch:

- das **Erstellungsdatum** („Erstellt am …“) im Kopf (PDF) bzw. als Kopfzeile (Excel), und
- die feste Kennzeichnung **„Erstellt durch Ordivis Platform“** im Fuß – diese bleibt immer erhalten.

Hinweis: Firmenname und Logo (ebenfalls unter *Branding*) erscheinen im Kopf der PDF-Berichte. CSV-Exporte bleiben reine Daten ohne Kopf/Fuß.

6 Berechtigungen

Berechtigung	Bedeutung
`reporting.read`	Reports und Feldkatalog sehen, Vorschau des Baukastens, Export
`reporting.create`	Reports anlegen und bearbeiten – **und die Vorschau freier SQL-Abfragen**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Berechtigung	Bedeutung
`reporting.manage`	Zeitpläne verwalten, alle Reports verwalten

Die Rolle **CIO** trägt alle drei Berechtigungen und ist damit die typische Führungs-Rolle für Auswertungen.

Warum der SQL-Weg mehr verlangt als das Leserecht. Eine freie SQL-Abfrage läuft unverändert gegen die Mandanten-Datenbank und damit an den Berechtigungen der einzelnen Module vorbei. Wer nur Berichte ansehen darf, käme über die Vorschau sonst an Identitäts-, Protokoll- und fremde Vorgangsdaten. Seit Version 2026.07.28.1912 verlangt die Vorschau im SQL-Modus deshalb `reporting.create` – dasselbe Recht, an dem gespeicherte SQL-Reports ohnehin hängen. Der Baukasten-Weg bleibt beim Leserecht. Fehlt das Recht, blendet der Client den SQL-Umschalter aus, statt den Wechsel erst anzubieten und dann abzuweisen.

Verwandte Themen

- **ID-27 – Modul Reporting & Dashboards** – Dashboards, Zeitpläne und Ausgabe der Berichte
- **ID-52 – Datenbank-Referenz** – vollständige Feld- und Tabellenreferenz
- **ID-51 – Datenbank-Leistung & Dimensionierung** – was aufwendige Abfragen für die Datenbank bedeuten
- **ID-44 – Rollen- & Rechte-Matrix** – Rechte, die auf Reports wirken

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · INFORMATIONSSICHERHEIT

Modul Informationssicherheit (ISMS)

ISMS nach BSI IT-Grundschutz und ISO/IEC 27001 in einem Modul

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-28 ISMS

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Informationssicherheit (ISMS)

Informationssicherheit ist mehr als ein Virenscanner. Es geht um die Frage, wie eine Organisation ihre Daten und Systeme planvoll schützt: Welche Informationen sind besonders schützenswert? Welche Gefahren drohen ihnen? Welche Maßnahmen sind getroffen, und wer ist dafür verantwortlich? Ohne System bleibt das Stückwerk – hier eine Regel, dort eine Vorkehrung, aber kein belastbares Gesamtbild und kein Nachweis gegenüber Prüfern.

Ein *ISMS* (Informationssicherheits-Managementsystem) ist der organisierte Rahmen für genau diese Fragen. Statt das Rad neu zu erfinden, stützt man sich dabei auf anerkannte Regelwerke. Zwei sind hier hinterlegt: der BSI IT-Grundschutz aus Deutschland und der internationale Standard ISO/IEC 27001. Beide liefern Kataloge von Anforderungen (sogenannte Controls oder Bausteine), die man Schritt für Schritt umsetzt und dokumentiert. Das Modul führt diese Kataloge, verknüpft sie mit den Schutzobjekten aus der CMDB und hält Umsetzungsstand, Risiken und Nachweise an einem Ort fest.

INHALT

1 Zweck & Nutzen
2 Frameworks & Kataloge
2.1 BSI Grundschutz++ automatisch aktuell halten
2.2 Referenzen des BSI – fertige Begründungen und Rahmenwerk-Zuordnungen
3 Schutzbedarf & Vererbung
4 Controls & Maßnahmen umsetzen
5 ISMS-Kontext & Geltungsbereich (ISO 27001, 4.2/4.3/5.2)
6 Risiken & Risikobehandlungsplan (ISO 27001, 6.1.3 / 8.3)
6.1 Risiken aus anderen Quellen
7 Erklärung zur Anwendbarkeit (SoA) (ISO 27001, 6.1.3 d)
8 ISMS-Betrieb & -Bewertung (ISO 27001, Klauseln 6.2/7.5/9/10)
9 Zertifizierungs-Readiness & Berichte (ISO 27001)
10 Export & Nachweise
11 Abgrenzung zu BCM & DR
12 Rollen & Rechte (Auszug)
13 Das dokumentierte ISMS – Anlage mit Vorlagen
Verwandte Themen

1 Zweck & Nutzen

- Strukturierte Umsetzung von Sicherheitsanforderungen.
- Nachweise für Audits und Zertifizierung.
- Verzahnung mit CMDB (Schutzobjekte) und Risiken.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2 Frameworks & Kataloge

Der **Sicherheitskatalog** zeigt die Einträge des gewählten Frameworks links als Liste und rechts das Detail zum markierten Eintrag. Welches Framework Sie sehen, steuert das Feld *Katalog* in der Kopfzeile – siehe Abbildung 1.

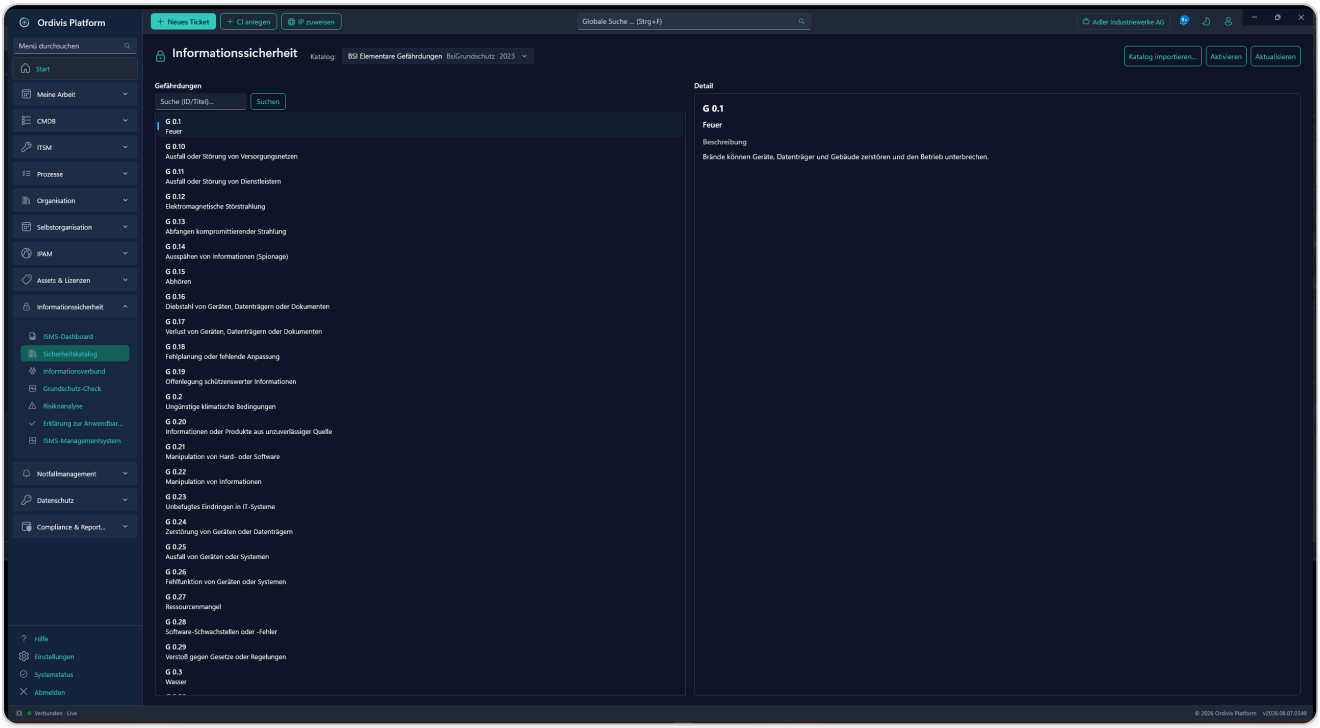

Abb. 1 – Sicherheitskatalog, hier mit dem Katalog *BSI Elementare Gefährdungen*: links die Einträge, rechts das Detail. Über *Katalog* wechseln Sie zu ISO/IEC 27001.

- **BSI IT-Grundschutz:** Bausteine, Anforderungen, Vererbungs-Engine.
- **ISO/IEC 27001:** 93 Annex-A-Controls + Klauseln 4–10 werden mit der Auslieferung geseedet und je Framework aktiviert (verdrängt BSI nicht).

> **Hinweis:** Ordivis Platform enthält **keinen ISO-Normtext**, sondern Titel und eigene Erläuterungen zu jedem Control. Den Normtext beziehen Sie separat beim Herausgeber.

2.1 BSI Grundschutz++ automatisch aktuell halten

Der Katalog **BSI Grundschutz++** kann automatisch mit der offiziellen BSI-Quelle abgeglichen werden:

- Ein täglicher Hintergrund-Job vergleicht den aktiven Katalog per Prüfsumme mit der veröffentlichten Fassung.
- Bei einer Abweichung wird der Katalog-Verantwortliche (`security.catalog.manage`) per In-App-Benachrichtigung und E-Mail informiert – **kein stiller Wechsel**.
- Die neue Fassung wird erst **nach Bestätigung** übernommen; eigene Anpassungen (Tailoring) bleiben erhalten. So bleibt der Katalogstand jederzeit revisionssicher nachvollziehbar.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Konfiguration:** `Security:BsiCatalog` (Auto-Prüfung an/aus, Quell-URL).

2.2 Referenzen des BSI – fertige Begründungen und Rahmenwerk-Zuordnungen

> 💻 **Client:** *Informationssicherheit* ▸ *Referenzen des BSI* · 🔑 `security.check.read` zum Lesen, `security.catalog.manage` zum Einlesen, `security.check.write` zum Bestätigen

Die BSI-Bibliothek enthält mehr als den Katalog. Zwei weitere Schichten nimmt Ordivis seit diesem Release **auf** – vorher hat es sie nur erzeugt:

Komponentendefinitionen beschreiben je Produkt, welche Anforderungen es umsetzt und **wie**. Für Keycloak, Netzarchitektur, Passwortrichtlinie, Lieferkettensicherheit und das GA-Lotse-Grundmodul liegen sie fertig vor. Wer eine dieser Komponenten betreibt, findet die Begründung zu den betroffenen Anforderungen bereits ausformuliert vor, statt sie selbst zu schreiben.

Rahmenwerk-Zuordnungen verbinden ISO 27001 Annex A und den früheren IT-Grundschutz mit Grundschutz++. Wer beides betreibt – für die öffentliche Verwaltung der Regelfall, sobald eine Zertifizierung dazukommt –, sieht damit, welche Umsetzung beide Rahmenwerke beantwortet.

🔒 **Pflichtreihenfolge** – ohne eingelesenen Katalog gibt es nichts, worauf sich die Verweise beziehen könnten.

1. Öffnen Sie *Informationssicherheit* ▸ *Referenzen des BSI*.
2. Wählen Sie unter *Komponentendefinitionen* eine Quelle und klicken Sie auf *Aus der BSI-Bibliothek laden*. Ohne Weg ins Internet laden Sie die Datei stattdessen als Datei hoch.
3. Wiederholen Sie das unter *Rahmenwerk-Zuordnungen*.
4. Geben Sie unten die Kennung einer Anforderung ein (z. B. `BER.1.4`) und klicken Sie auf *Suchen*.
5. Prüfen Sie den vorgeschlagenen Text und klicken Sie auf *Bestätigt*, wenn er für Ihre Installation zutrifft.

✅ **Ergebnis:** Der bestätigte Text steht als Umsetzung dieser Institution zur Verfügung. Solange er nicht bestätigt ist, bleibt er sichtbar als *Vorschlag* markiert.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

⚠️ **Ein Vorschlag ist keine Umsetzung.** Das BSI sagt ausdrücklich, dass Komponentendefinitionen Referenzen sind und „nicht das Handeln der für die Informationssicherheit verantwortlichen Stelle" ersetzen. Ordivis übernimmt deshalb nichts von selbst.

⚠️ **Nicht jeder Verweis passt zu Ihrem Katalogstand.** Die oberen Schichten des BSI verweisen auf den Stand, gegen den sie geschrieben wurden. Was sich in Ihrem Katalog nicht wiederfindet, wird **gezählt und angezeigt** statt weggelassen – eine Meldung „vollständig", die einen Teil verschweigt, wäre schlimmer als die ehrliche Zahl. Gemessen an der Fassung vom 13.08.2026 lösen sich 34 von 43 Keycloak-Verweisen auf und 293 von 365 Zuordnungszielen.

⚠️ **Der Beziehungstyp gehört gelesen.** Nur *gleichwertig* trägt eine Umsetzung unverändert von einem Rahmenwerk ins andere. Bei *deckt nur einen Teil ab* und *überschneidet sich teilweise* bleibt Arbeit übrig. Ordivis zeigt den Typ im Klartext und macht daraus **kein** Häkchen – in der ISO-Sammlung sind nur 16 von 96 Zuordnungen gleichwertig.

⚠️ **Entwurfsstand:** Trägt eine Komponentendefinition „Entwurf" im Titel, steht das an der Oberfläche. Die Keycloak-Definition ist im August 2026 genau so eine.

3 Schutzbedarf & Vererbung

Sie definieren den **Schutzbedarf** (Vertraulichkeit, Integrität, Verfügbarkeit) an Objekten. Eine **Vererbungs-Engine** propagiert Schutzbedarf entlang von Abhängigkeiten (z. B. Dienst → Server).

4 Controls & Maßnahmen umsetzen

💻 **Client:** *Informationssicherheit* ▸ *Sicherheitskatalog* · 🌐 **Portal:** *Informationssicherheit* ▸ *ISMS-Arbeitsplatz* · 🔑 `security.report.read` zum Lesen, `security.manage` zum Pflegen

🔒 **Pflichtreihenfolge** — das **Framework** bestimmt, welche Controls und Bausteine überhaupt angeboten werden. BSI-Bausteine und ISO-Controls sind nicht dieselbe Liste; ein Wechsel nach dem Erfassen ordnet die Nachweise nicht um.

1. Wählen Sie das **Framework** (BSI oder ISO 27001).
2. Öffnen Sie einen **Control/Baustein**.
3. Erfassen Sie **Umsetzungsstatus**, Verantwortliche und Nachweise.
4. Verknüpfen Sie die betroffenen **CIs/Assets**.
5. Speichern Sie.

✅ **Ergebnis:** Der Umsetzungsstand fließt in die Erklärung zur Anwendbarkeit (SoA) und in den Grundschutz-Check ein – Sie erfassen ihn einmal und weisen ihn mehrfach nach.

Tipp: Ein Nachweis ohne Datei ist im Audit keiner. Hängen Sie das Dokument an, das die Umsetzung belegt, statt sie nur zu behaupten.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5 ISMS-Kontext & Geltungsbereich (ISO 27001, 4.2/4.3/5.2)

Jeder **Informationsverbund** trägt seinen ISO-Kontext als dokumentierte Information: die **Geltungsbereichs-Erklärung** (Grenzen und Anwendbarkeit, 4.3), die **interessierten Parteien** samt Anforderungen (4.2) und die **Informationssicherheits-Leitlinie** (5.2). Sie pflegen diese Felder über die Schaltfläche **„ISMS-Kontext"** auf der Verbund-Seite – sie sind die Grundlage für Scope-Statement und Zertifizierungsnachweis.

6 Risiken & Risikobehandlungsplan (ISO 27001, 6.1.3 / 8.3)

Erfassen und bewerten Sie **Risiken** (Eintrittswahrscheinlichkeit × Auswirkung), hinterlegen Maßnahmen und verfolgen deren Wirkung. Über **„Behandlungsplan"** legen Sie je Risiko die Behandlungsstrategie fest (Vermeiden / Reduzieren / Übertragen / Akzeptieren) und **verknüpfen die umsetzenden Annex-A-Controls**.

> **Wichtig:** Bei der Strategie **„Reduzieren"** muss mindestens ein Annex-A-Control verknüpft sein – sonst gilt das Risiko als unbehandelt, und die Anwendung lehnt das Speichern ab (ISO 27001, 6.1.3). Die formale **Restrisiko-Akzeptanz** durch die Leitung wird gesondert dokumentiert.

6.1 Risiken aus anderen Quellen

Zwei Analysen entstehen nicht von Hand, sondern werden befüllt. Sie tragen sprechende Namen und liegen im jeweiligen Informationsverbund:

Analyse	Quelle	Bewertung
Kontinuitäts-Gaps (BCM)	offene Wiederanlauf-Unterdeckungen (RTA > RTO) aus dem BCM-Modul	Schadenshöhe aus der BIA
Schwachstellen (Scan)	gemeldete CVE-/WID-Befunde eines angebundenen Werkzeugs	Schadenshöhe aus dem CVSS-Basiswert

Beide Wege sind **idempotent**: Derselbe Befund am selben Objekt aktualisiert sein Risiko, statt ein zweites anzulegen. Dieselbe Schwachstelle auf zwei Rechnern ergibt dagegen **zwei** Risiken – sie werden getrennt behandelt und getrennt geschlossen.

Die Abbildung CVSS → Schadenshöhe folgt den CVSS-v3-Schweregradbändern (≥ 9,0 → 5; ≥ 7,0 → 4; ≥ 4,0 → 3; > 0 → 2). **Fehlt der Wert, wird 3 („mittel") angesetzt** – ein Befund ohne Bewertung ist unbekannt, nicht harmlos.

> **Was NICHT übertragen wird:** Nur die Referenz kommt an – Kennung, Titel, betroffenes Bauteil, CVSS und Fundzeitpunkt. Keine Exploit-Details. Ein Betriebsdatenbestand ist kein Ort für Anleitungen.

Ein Befund wird nur dann an ein **Zielobjekt** gehängt, wenn dieses Objekt in diesem Verbund tatsächlich im Betrachtungsbereich liegt. Sonst bleibt er ein Risiko der Analyse mit dem Rechnernamen im Text – ein Zielobjekt aus einem fremden Verbund zu verknüpfen wäre fachlich falsch.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

7 Erklärung zur Anwendbarkeit (SoA) (ISO 27001, 6.1.3 d)

Das **Statement of Applicability** ist das zentrale Zertifizierungs-Artefakt. Die SoA-Matrix führt **alle 93 Annex-A-Controls** und wird automatisch vorbefüllt (Standard: anwendbar, Umsetzungsstatus aus dem Grundschutz-Check abgeleitet). Je Control halten Sie fest: **anwendbar ja/nein**, **Begründung** für Ein- oder Ausschluss, Umsetzungsstatus und Verantwortliche.

> **Wichtig:** Ein **Ausschluss ohne Begründung** ist unzulässig und wird abgewiesen. Die vollständige SoA lässt sich als **versioniertes PDF** exportieren.

8 ISMS-Betrieb & -Bewertung (ISO 27001, Klauseln 6.2/7.5/9/10)

Die Seite **„ISMS-Managementsystem"** bündelt die Management-System-Schicht, die ISO von einer reinen Control-Checkliste unterscheidet:

- **Ziele & Kennzahlen (6.2/9.1):** messbare ISMS-Ziele mit KPI, Zielwert und Fortschritt.
- **Internes Audit (9.2):** Audit-Findings (Nichtkonformität / Beobachtung / Verbesserungspotential); ein Finding lässt sich in eine Nichtkonformität überführen.
- **Managementbewertung (9.3):** Inputs (Audit-Ergebnisse, Risiken, Ziele) und Beschlüsse/Outputs, mit **Freigabe durch die Leitung**.
- **Nichtkonformitäten & Korrekturmaßnahmen (10.1/10.2):** Ursachenanalyse → Korrekturmaßnahme → **Wirksamkeitsprüfung**. Eine Nichtkonformität kann erst **nach dokumentierter Wirksamkeitsprüfung** geschlossen werden.
- **Dokumentenlenkung (7.5):** Register der verbindlich dokumentierten Information mit Version, Owner und Freigabestatus.

9 Zertifizierungs-Readiness & Berichte (ISO 27001)

Das **Readiness-Dashboard** zeigt eine **„Stage-1-bereit"-Ampel**: SoA-Vollständigkeit, Abdeckung der Management-System-Klauseln 4–10, offene Nichtkonformitäten und die Pflichtdokumenten-Checkliste – inklusive Hinweis auf offene Lücken. Als PDF erzeugen Sie: **SoA**, **Risikobehandlungsplan**, **Internes-Audit-Bericht** und **Management-Review-Protokoll**.

10 Export & Nachweise

Der **OSCAL-Export** stellt Kontroll- und Umsetzungsdaten in einem maschinenlesbaren Standardformat bereit. Profile bündeln relevante Controls je Anwendungsbereich.

11 Abgrenzung zu BCM & DR

Das ISMS betrachtet **Sicherheit** (Schutzziele, Controls, Risiken). **Geschäftsfortführung** und **Wiederanlauf** behandeln die Module BCM (ID-29) und DR-Planung (ID-30) – eng verzahnt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

12 Rollen & Rechte (Auszug)

Recht	Erlaubt
`security.check.read`	ISMS-Daten ansehen (Katalog, Verbünde, SoA, Readiness)
`security.scope.manage`	Informationsverbünde und ISMS-Kontext pflegen
`security.assessment.manage`	Schutzbedarf und SoA-Einträge pflegen
`security.risk.manage`	Risiken und Risikobehandlungsplan pflegen
`security.audit.manage`	Grundschutz-Check, ISMS-Ziele, Audits, Nichtkonformitäten, Dokumente
`security.risk.accept`	Restrisiko akzeptieren, Managementbewertung freigeben
`security.report.generate`	Berichte (SoA, RTP, Audit, Review) erzeugen
`security.catalog.manage`	Kataloge und Frameworks verwalten

13 Das dokumentierte ISMS – Anlage mit Vorlagen

Ein zertifizierungsfähiges ISMS braucht neben dem Modul auch eine Reihe **freigegebener Dokumente** (Leitlinie, Scope, Risikoverfahren, Erklärung zur Anwendbarkeit, Auditberichte …). Welche Dokumente die ISO/IEC 27001:2022 verlangt und wie sie aussehen, beschreibt die eigenständige **Anlage *ID-28a ISMS-Dokumentation*** – mit ausformulierten Vorlagen für jedes Pflichtdokument im Ordivis-Corporate-Design.

Verwandte Themen

- **ID-28a – Anlage: ISMS-Dokumentation nach ISO/IEC 27001** – die ausformulierte ISMS-Dokumentation
- **ID-29 – Modul Business Continuity Management** – Fortführung des Geschäftsbetriebs
- **ID-30 – Modul Notfall- & Wiederanlaufplanung (DR)** – technischer Wiederanlauf nach einem Ausfall
- **ID-32 – Modul Datenschutz (Art. 30 DSGVO)** – Schnittmenge mit dem Datenschutz
- **ID-21 – Modul CMDB** – Schutzbedarf an den Configuration Items

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

ANLAGE · INFORMATIONSSICHERHEIT

Anlage: ISMS-Dokumentation nach ISO/IEC 27001

Das dokumentierte ISMS nach ISO/IEC 27001:2022 – Lenkung und Vorlagen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-28a ISMS-DOK

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Anlage: ISMS-Dokumentation nach ISO/IEC 27001

Ein Informationssicherheits-Managementsystem (ISMS) lebt nicht von guten Absichten, sondern von nachweisbaren Festlegungen. Die ISO/IEC 27001:2022 verlangt an mehreren Stellen ausdrücklich *„dokumentierte Information"* – also schriftlich festgehaltene Regeln, Verfahren und Nachweise. Wer eine Zertifizierung anstrebt oder gegenüber Auftraggebern, Aufsichtsbehörden und Versicherern belastbar sein will, kommt an dieser Dokumentation nicht vorbei.

Diese Anlage ist bewusst **eigenständig** gehalten: Sie beschreibt ausschließlich das dokumentierte ISMS und stellt für jedes Pflichtdokument eine ausformulierte **Vorlage** bereit. Das Modul Informationssicherheit (siehe Kapitel *ID-28 ISMS*) verwaltet die technische Seite – Kataloge, Schutzobjekte, Risiken, Prüfungen. Diese Anlage liefert das Papier drumherum, das ein Auditor sehen möchte.

> **Wichtig.** Ordivis Platform liefert die Kataloge (ISO/IEC 27001 Annex A und BSI IT-Grundschutz) und die Werkzeuge zur Modellierung, Risikobewertung und Prüfung bereits mit. Die hier abgedruckten Vorlagen sind **Textbausteine für Ihre organisatorischen Dokumente** – sie ersetzen keine Rechtsberatung und keinen Auditor. Passen Sie jede Vorlage an Ihre Organisation an, bevor Sie sie in Kraft setzen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

1 Wie die Dokumentation aufgebaut ist
2 Das Dokumentenregister – was die Norm verlangt
3 Zuständigkeiten – wer erstellt und pflegt was
4 Lebenszyklus & Lenkung dokumentierter Information
5 Die Vorlagen
5.0 Vorlage 0 – Lenkungskopf (für jedes Dokument)
5.1 D-01 – Anwendungsbereich des ISMS (Scope)
5.2 D-02 – Informationssicherheitsleitlinie
5.3 D-03 – Rollen, Verantwortlichkeiten & Befugnisse
5.4 D-04 – Verfahren zur Risikobeurteilung
5.5 D-05 – Verfahren zur Risikobehandlung
5.6 D-06 – Erklärung zur Anwendbarkeit (Statement of Applicability, SoA)
5.7 D-07 – Risikobehandlungsplan
5.8 D-08 – Informationssicherheitsziele
5.9 D-09 – Nachweis der Kompetenz & Awareness
5.10 D-10 – Risikobeurteilungsbericht
5.11 D-11 – Nachweis der Wirksamkeitsmessung (Monitoring)
5.12 D-12 – Internes Auditprogramm & Auditbericht
5.13 D-13 – Managementbewertung (Management Review)
5.14 D-14 – Nichtkonformitäten & Korrekturmaßnahmen
5.15 D-15 – Zugriffssteuerungsrichtlinie
5.16 D-16 – Richtlinie zur akzeptablen Nutzung
5.17 D-17 – Lieferanten- & Cloud-Sicherheitsrichtlinie
5.18 D-18 – Kryptografie- & Schlüsselrichtlinie
5.19 D-19 – Incident-Management-Verfahren
5.20 D-20 – Betriebskontinuität (Bezug zum BCM)
6 Startpaket – die Reihenfolge der Einführung
7 Verweise
Verwandte Themen

1 Wie die Dokumentation aufgebaut ist

Die ISMS-Dokumentation folgt einer bewährten **Vier-Ebenen-Pyramide**. Von oben nach unten wird sie konkreter und ändert sich häufiger:

Ebene	Art	Beispiele	Ändert sich
1	**Politik** (das *Warum*)	Informationssicherheitsleitlinie	selten
2	**Richtlinien** (das *Was*)	Zugriffs-, Krypto-, Lieferanten-Richtlinie	gelegentlich
3	**Verfahren & Prozesse** (das *Wie*)	Risikoverfahren, Incident-Prozess, Change-Prozess	regelmäßig
4	**Nachweise** (der *Beweis*)	Risikoregister, Auditberichte, Schulungsnachweise	laufend

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Tipp. Halten Sie die oberen Ebenen kurz und stabil, die unteren detailliert und lebendig. Eine Leitlinie über zwei Seiten, die zehn Jahre trägt, ist mehr wert als ein 40-seitiges Werk, das niemand liest.

Jedes Dokument trägt einen **Lenkungskopf** (siehe Vorlage 0), damit Version, Freigabe und Gültigkeit jederzeit nachvollziehbar sind. Das ist keine Förmelei: Die Norm verlangt in Abschnitt 7.5, dass dokumentierte Information *gekennzeichnet, geprüft, freigegeben und versioniert* ist.

2 Das Dokumentenregister – was die Norm verlangt

Die folgende Tabelle listet die dokumentierte Information, die ISO/IEC 27001:2022 fordert (Spalte *Pflicht*) sowie die in der Praxis üblichen ergänzenden Dokumente. Die Spalte *Klausel* verweist auf den Normabschnitt, *Vorlage* auf den Abschnitt dieser Anlage.

Nr.	Dokument	Klausel	Pflicht	Vorlage
D-01	Anwendungsbereich des ISMS (Scope)	4.3	ja	§5.1
D-02	Informationssicherheitsleitlinie	5.2	ja	§5.2
D-03	Rollen, Verantwortlichkeiten & Befugnisse	5.3	ja	§5.3
D-04	Verfahren zur Risikobeurteilung	6.1.2	ja	§5.4
D-05	Verfahren zur Risikobehandlung	6.1.3	ja	§5.5
D-06	Erklärung zur Anwendbarkeit (SoA)	6.1.3 d)	ja	§5.6
D-07	Risikobehandlungsplan	6.1.3 e)	ja	§5.7
D-08	Informationssicherheitsziele	6.2	ja	§5.8
D-09	Nachweis der Kompetenz	7.2	ja	§5.9
D-10	Risikobeurteilungsbericht (Ergebnisse)	8.2	ja	§5.10
D-11	Nachweis der Wirksamkeitsmessung	9.1	ja	§5.11
D-12	Internes Auditprogramm & -berichte	9.2	ja	§5.12
D-13	Managementbewertung (Protokoll)	9.3	ja	§5.13
D-14	Nichtkonformitäten & Korrekturmaßnahmen	10.2	ja	§5.14
D-15	Zugriffssteuerungsrichtlinie	A.5.15	empfohlen	§5.15
D-16	Richtlinie akzeptable Nutzung	A.5.10	empfohlen	§5.16
D-17	Lieferanten- & Cloud-Sicherheitsrichtlinie	A.5.19–5.23	empfohlen	§5.17
D-18	Kryptografie- & Schlüsselrichtlinie	A.8.24	empfohlen	§5.18
D-19	Incident-Management-Verfahren	A.5.24–5.28	empfohlen	§5.19
D-20	Betriebskontinuität (Bezug zum BCM)	A.5.29–5.30	empfohlen	§5.20

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Achtung. Die Erklärung zur Anwendbarkeit (SoA, D-06) ist das Herzstück jeder Zertifizierung. Sie muss **alle 93 Controls des Annex A** aufführen und je Control begründen, ob es angewandt wird und warum (bzw. warum nicht). Ordivis Platform erzeugt das Grundgerüst dieser Liste aus dem aktiven ISO-Katalog – die Begründungen tragen Sie ein.

3 Zuständigkeiten – wer erstellt und pflegt was

Dokument(e)	Erstellung	Freigabe	Pflege
D-01, D-02, D-08	ISB*	oberste Leitung	jährlich
D-03	ISB	oberste Leitung	bei Änderung
D-04, D-05, D-07, D-10	ISB / Risikoverantwortliche	ISB	pro Zyklus
D-06 (SoA)	ISB	oberste Leitung	pro Zyklus
D-09, D-11	ISB / Personal	ISB	laufend
D-12, D-13, D-14	interne Revision / ISB	Leitung	pro Ereignis
D-15 … D-20	Fachverantwortliche	ISB	bei Änderung

* ISB = Informationssicherheitsbeauftragte(r). In Ordivis Platform existiert dafür die Systemrolle **ISB** mit allen `security.*`-Berechtigungen.

4 Lebenszyklus & Lenkung dokumentierter Information

1. **Entwurf** → Autor erstellt anhand der Vorlage, Status *Entwurf*.
2. **Prüfung** → fachliche und formale Prüfung, Kommentare eingearbeitet.
3. **Freigabe** → Freigabe durch die zuständige Stelle (siehe §3), Status *Gültig*, Version erhöht.
4. **Verteilung** → Bekanntgabe an den betroffenen Personenkreis.
5. **Überprüfung** → im festgelegten Turnus (i. d. R. jährlich) auf Aktualität geprüft.
6. **Außerkraftsetzung** → veraltete Fassungen als *Archiviert* gekennzeichnet, aber revisionssicher aufbewahrt.

Tipp. Bilden Sie ISMS-Dokumente als Configuration Items der CI-Klasse **Dokument/Richtlinie** in der CMDB ab (Governance-&-Compliance-Klassen, BSI-07). So haben Sie Version, Eigentümer und Wiedervorlage an einem Ort und können Dokumente mit Schutzobjekten verknüpfen.

5 Die Vorlagen

Die folgenden Vorlagen sind vollständig ausformuliert. Ersetzen Sie die in **«Spitzklammern»** gesetzten Platzhalter durch Ihre Angaben. Textteile in *kursiv* sind Ausfüllhinweise und werden vor Inkraftsetzung entfernt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5.0 Vorlage 0 – Lenkungskopf (für jedes Dokument)

Jedes ISMS-Dokument beginnt mit diesem Kopf:

Feld	Inhalt
Dokumenttitel	«z. B. Informationssicherheitsleitlinie»
Dokumentnummer	«D-02»
Version	«1.0»
Klassifizierung	«intern / vertraulich»
Eigentümer	«Rolle/Person»
Erstellt von / am	«Name», «TT.MM.JJJJ»
Geprüft von / am	«Name», «TT.MM.JJJJ»
Freigegeben von / am	«Leitung», «TT.MM.JJJJ»
Gültig ab	«TT.MM.JJJJ»
Nächste Überprüfung	«TT.MM.JJJJ»

Änderungshistorie

Version	Datum	Autor	Änderung
1.0	«TT.MM.JJJJ»	«Name»	Erstfassung

5.1 D-01 – Anwendungsbereich des ISMS (Scope)

Zweck: Legt fest, welche Teile der Organisation, welche Standorte, Prozesse und Werte das ISMS abdeckt (Klausel 4.3). Der Scope ist die Grenze, innerhalb derer alle weiteren Festlegungen gelten.

1. Organisation. Das ISMS gilt für «Organisation/Gesellschaft», nachfolgend „die Organisation".

2. Eingeschlossene Bereiche. Vom ISMS erfasst sind: - Standorte: «z. B. Hauptsitz Musterstadt, Rechenzentrum X» - Organisationseinheiten: «z. B. IT-Betrieb, Entwicklung, Kundenservice» - Geschäftsprozesse: «z. B. Betrieb der SaaS-Plattform, Kundendatenverarbeitung» - Informationswerte: «z. B. Kundendaten, Quellcode, Betriebsdokumentation»

3. Ausgeschlossene Bereiche und Begründung. Nicht erfasst sind «Bereich» *(Begründung: «…»)*.

4. Schnittstellen & Abhängigkeiten. Externe Abhängigkeiten mit Sicherheitsrelevanz: «z. B. Cloud-Provider, Managed-Service-Dienstleister, Zahlungsdienstleister».

5. Interessierte Parteien und deren Anforderungen (Klausel 4.2): «Kunden (Vertraulichkeit), Aufsicht (DSGVO/NIS-2), Versicherer (Mindeststandards)».

> **Wichtig.** Der Scope muss zum Zertifikat passen: Nur was hier steht, wird geprüft und zertifiziert. Ein zu weiter Scope überfordert, ein zu enger entwertet das Zertifikat.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5.2 D-02 – Informationssicherheitsleitlinie

Zweck: Das oberste Bekenntnis der Leitung zur Informationssicherheit (Klausel 5.2). Kurz, verbindlich, von der Geschäftsführung unterschrieben.

Präambel. Informationen sind ein wesentlicher Wert der «Organisation». Ihr Schutz hinsichtlich **Vertraulichkeit, Integrität und Verfügbarkeit** ist Grundlage unseres Geschäfts und unserer Vertrauenswürdigkeit. Die Leitung bekennt sich zu einem systematischen Informationssicherheits-Management nach ISO/IEC 27001.

Grundsätze. 1. Informationssicherheit ist Leitungsaufgabe und Teil aller Geschäftsprozesse. 2. Wir beurteilen Risiken systematisch und behandeln sie nach festgelegten Kriterien. 3. Gesetzliche, vertragliche und regulatorische Anforderungen werden eingehalten. 4. Jede/r Beschäftigte trägt Verantwortung und wird geschult. 5. Sicherheitsvorfälle werden gemeldet, bearbeitet und ausgewertet. 6. Wir verbessern das ISMS fortlaufend (PDCA).

Ziele. Konkrete, messbare Ziele sind im Dokument D-08 festgelegt.

Verbindlichkeit. Diese Leitlinie ist für alle Beschäftigten und für Dritte im Geltungsbereich verbindlich. Verstöße werden gemäß «Regelung» geahndet.

Freigabe. «Ort, Datum» – «Name, Funktion (Geschäftsführung)» – Unterschrift: ___

5.3 D-03 – Rollen, Verantwortlichkeiten & Befugnisse

Zweck: Weist die sicherheitsrelevanten Rollen zu (Klausel 5.3).

Rolle	Verantwortung	Besetzt mit
Oberste Leitung	Gesamtverantwortung, Ressourcen, Freigaben	«Name»
Informationssicherheitsbeauftragte(r) (ISB)	Betrieb & Steuerung des ISMS	«Name»
Risikoeigentümer	Bewertung/Behandlung „ihrer" Risiken	«Bereich»
IT-Betrieb	Umsetzung technischer Maßnahmen	«Team»
Datenschutzbeauftragte(r)	Schnittstelle DSGVO	«Name»
Alle Beschäftigten	Einhaltung, Meldung von Vorfällen	–

Tipp. In Ordivis Platform bildet die Rolle **ISB** die/den Informationssicherheitsbeauftragte(n) technisch ab. Weisen Sie sie in der Benutzerverwaltung genau den Personen zu, die hier benannt sind.

5.4 D-04 – Verfahren zur Risikobeurteilung

Zweck: Legt fest, wie Risiken identifiziert, analysiert und bewertet werden (Klausel 6.1.2) – reproduzierbar und vergleichbar.

1. Methode. Risiken werden je Schutzobjekt aus **Gefährdung** × **Schwachstelle** abgeleitet und nach **Eintrittshäufigkeit** und **Schadenshöhe** bewertet.

2. Bewertungsskala. Je Dimension 1–5; das Risiko ergibt sich als Produkt (1–25):

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Risikowert	Kategorie	Bedeutung
1–4	akzeptabel	keine weitere Maßnahme nötig
5–12	tragbar	Maßnahmen prüfen
13–25	kritisch	Behandlung verbindlich, Freigabe durch Leitung

3. Gefährdungsbasis. Grundlage sind die **47 elementaren Gefährdungen des BSI** sowie der **Enterprise-Gefährdungskatalog** (moderne Risiken: Supply-Chain, Cloud, KI, Regulatorik). Beide sind in Ordivis Platform hinterlegt.

4. Ablauf. Schutzbedarf feststellen → Gefährdungen zuordnen → bewerten → dokumentieren → Ergebnis in Bericht D-10.

5. Turnus. Mindestens jährlich sowie bei wesentlichen Änderungen.

> **Wichtig.** Ordivis Platform setzt genau dieses Verfahren um: 5×5-Bewertung, Kategorien, Heat-Map und automatische Benachrichtigung bei kritischem Restrisiko. Dieses Dokument beschreibt die *Regel*, das Modul liefert die *Durchführung*.

5.5 D-05 – Verfahren zur Risikobehandlung

Zweck: Wie mit bewerteten Risiken umgegangen wird (Klausel 6.1.3).

Strategien. Für jedes nicht akzeptable Risiko wird eine Strategie gewählt: - **Reduzieren** – Maßnahmen (Controls) umsetzen. - **Vermeiden** – die risikobehaftete Tätigkeit einstellen. - **Übertragen** – z. B. Versicherung, Auslagerung mit vertraglicher Absicherung. - **Akzeptieren** – bewusste, dokumentierte und **freigegebene** Hinnahme des Restrisikos.

Restrisiko. Nach Behandlung wird das Restrisiko erneut bewertet. Kritische Restrisiken bedürfen der **ausdrücklichen Freigabe durch die oberste Leitung**.

Auswahl der Maßnahmen. Maßnahmen werden aus dem Annex A (bzw. BSI-Bausteinen) abgeleitet; die Auswahl wird in der SoA (D-06) begründet und im Risikobehandlungsplan (D-07) terminiert.

5.6 D-06 – Erklärung zur Anwendbarkeit (Statement of Applicability, SoA)

Zweck: Führt alle 93 Annex-A-Controls auf und begründet je Control Anwendung/Ausschluss (Klausel 6.1.3 d). Das zentrale Prüfdokument.

Aufbau je Zeile:

Control	Titel	Anwendbar	Umsetzungs-status	Begründung / Ausschlussgrund	Nachweis
A.5.1	Informationssicherheits-richtlinien	ja	umgesetzt	Leitlinie D-02 in Kraft	D-02
A.5.7	Threat Intelligence	ja	in Umsetzung	Prozess im Aufbau	«…»

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Control	Titel	Anwendbar	Umsetzungs-status	Begründung / Ausschlussgrund	Nachweis
A.8.23	Web-Filter	nein	–	keine Web-Nutzung im Scope	–
…	*(alle 93 Controls)*				

Achtung. Kein Annex-A-Control darf fehlen. Ausschlüsse sind zulässig, müssen aber **sachlich begründet** sein („nicht anwendbar, weil …"). „Vergessen" ist kein Ausschlussgrund.

Tipp. Ordivis Platform erzeugt das SoA-Grundgerüst aus dem aktiven ISO-Katalog und dem Modellierungsstand. Exportieren Sie die Liste, ergänzen Sie Begründung und Nachweis, und legen Sie sie als freigegebenes Dokument ab.

5.7 D-07 – Risikobehandlungsplan

Zweck: Der terminierte Maßnahmenplan (Klausel 6.1.3 e).

Nr.	Risiko	Maßnahme (Control)	Verantwortlich	Frist	Status	Restrisiko
1	«R-…»	«A.8.x – …»	«Name»	«TT.MM.JJJJ»	offen	«Kat.»

Wichtig. Ordivis Platform führt diesen Plan als *Grundschutz-Check*-Maßnahmen bzw. Risikobehandlung im Modul – Status *offen → in Arbeit → umgesetzt → verifiziert*. Der ausgedruckte Maßnahmenplan-Bericht aus Ordivis Platform ist der Nachweis zu diesem Dokument.

5.8 D-08 – Informationssicherheitsziele

Zweck: Messbare Ziele je Planungszeitraum (Klausel 6.2).

Ziel	Kennzahl	Ausgangswert	Zielwert	Frist	Verantwortlich
Phishing-Resistenz erhöhen	Klickrate Testphishing	«18 %»	«< 5 %»	«Q4»	ISB
Patch-Rückstand senken	offene kritische CVEs	«30»	«0»	«laufend»	IT-Betrieb
Awareness	geschulte MA	«60 %»	«100 %»	«Q3»	Personal

Ziele sind **spezifisch, messbar, terminiert** und werden in der Managementbewertung (D-13) überprüft.

5.9 D-09 – Nachweis der Kompetenz & Awareness

Zweck: Beleg, dass Beteiligte kompetent und sensibilisiert sind (Klausel 7.2/7.3).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Person/Rolle	Erforderliche Kompetenz	Nachweis	Datum	Nächste Auffrischung
ISB	ISO-27001-Grundlagen	«Zertifikat»	«…»	«…»
Alle MA	Security-Awareness	Schulungsteilnahme	«…»	jährlich

5.10 D-10 – Risikobeurteilungsbericht

Zweck: Dokumentiertes Ergebnis der Risikobeurteilung (Klausel 8.2).

Enthält: Stichtag, betrachteter Scope, angewandtes Verfahren (Verweis D-04), **Risikoregister** (alle Risiken mit Brutto-/Restwert und Kategorie), Heat-Map und Gesamteinschätzung der Risikolage.

> **Tipp.** Das **Risikoregister-PDF** aus Ordivis Platform (farbcodiert nach Kategorie, mit Heat-Map) erfüllt diesen Zweck unmittelbar. Ergänzen Sie ein kurzes Vorwort und die Freigabe.

5.11 D-11 – Nachweis der Wirksamkeitsmessung (Monitoring)

Zweck: Beleg, dass die Wirksamkeit gemessen wird (Klausel 9.1).

Was wird gemessen	Methode	Turnus	Ergebnis	Verantwortlich
Umsetzungsgrad Controls	Grundschutz-Check-Fortschritt	quartalsweise	«xx %»	ISB
Risikolage	Anzahl kritischer Risiken	monatlich	«n»	ISB
Vorfälle	Incident-Statistik	monatlich	«n»	IT-Betrieb

> **Tipp.** Das **ISMS-Dashboard** in Ordivis Platform liefert Schutzbedarfsverteilung, Check-Fortschritt und Risikolage auf einen Blick – die laufende Datengrundlage für diesen Nachweis.

5.12 D-12 – Internes Auditprogramm & Auditbericht

Zweck: Planung und Ergebnis interner Audits (Klausel 9.2).

Auditprogramm (Jahresplan)

Audit	Bereich/Controls	Termin	Auditor	Status
A-1	A.5 organisatorisch	«Q1»	«Name (unabhängig)»	geplant

Auditbericht (je Audit): Ziel & Umfang, Kriterien, Feststellungen (Konformitäten/Abweichungen), Empfehlungen, Fazit. Abweichungen fließen in D-14 ein.

> **Wichtig.** Der Auditor muss vom geprüften Bereich **unabhängig** sein. In kleinen Organisationen ist ein externer Dienstleister oft die praktikabelste Lösung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5.13 D-13 – Managementbewertung (Management Review)

Zweck: Regelmäßige Bewertung des ISMS durch die Leitung (Klausel 9.3). Mindestens jährlich.

Pflicht-Eingaben (Agenda): 1. Status offener Maßnahmen aus der letzten Bewertung 2. Änderungen interner/externer Themen 3. Rückmeldungen zur Sicherheitsleistung (Kennzahlen D-11) 4. Ergebnisse von Audits (D-12) 5. Erfüllungsgrad der Ziele (D-08) 6. Vorfälle & Nichtkonformitäten (D-14) 7. Rückmeldungen interessierter Parteien 8. Verbesserungsmöglichkeiten

Pflicht-Ergebnisse (Beschlüsse): Verbesserungsentscheidungen, Ressourcenbedarf, angepasste Ziele.

Protokoll: Teilnehmer, Datum, behandelte Punkte, **Beschlüsse mit Verantwortlichen und Fristen**.

5.14 D-14 – Nichtkonformitäten & Korrekturmaßnahmen

Zweck: Umgang mit Abweichungen (Klausel 10.2).

Nr.	Abweichung	Quelle	Sofortmaßnahme	Ursachenanalyse	Korrekturmaßnahme	Frist	Wirksamkeit geprüft
N-1	«…»	Audit A-1	«…»	«…»	«…»	«…»	«ja/nein»

Wichtig. Nicht nur den Symptom beheben, sondern die **Ursache** – sonst kehrt die Abweichung zurück. Die Wirksamkeit der Korrektur ist zu prüfen und zu dokumentieren.

5.15 D-15 – Zugriffssteuerungsrichtlinie

Bezug: A.5.15/A.5.18/A.8.2 f. Regelt, wer worauf zugreifen darf.

Kernfestlegungen: **Need-to-know** und **least privilege**; rollenbasierte Rechtevergabe; Genehmigung und Entzug von Zugängen; regelmäßige **Rezertifizierung** der Rechte; besondere Kontrolle **privilegierter** Konten; MFA für «Systeme». Verantwortlich: «Rolle». Überprüfung: jährlich.

5.16 D-16 – Richtlinie zur akzeptablen Nutzung

Bezug: A.5.10. Was Nutzer mit Informationen und Mitteln dürfen – und was nicht.

Regelt u. a.: dienstliche vs. private Nutzung, Umgang mit E-Mail/Internet, Clean-Desk/Clear-Screen, Umgang mit mobilen Geräten und Wechseldatenträgern, Verbot ungenehmigter Cloud-/KI-Tools (Schatten-IT), Meldepflicht bei Vorfällen. Von jeder/m Beschäftigten zu **bestätigen**.

5.17 D-17 – Lieferanten- & Cloud-Sicherheitsrichtlinie

Bezug: A.5.19–A.5.23. Sicherheit in der Lieferkette und bei Cloud-Diensten.

Festlegungen: Sicherheitsanforderungen in Verträgen/AVV; Bewertung von Dienstleistern vor Beauftragung; Umgang mit **Nth-Party**-Abhängigkeiten; Regelung der geteilten Verantwortung (Shared Responsibility) bei Cloud; Exit-/Rückführungsstrategie gegen Vendor-Lock-in; regelmäßige Überprüfung der Dienstleister.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5.18 D-18 – Kryptografie- & Schlüsselrichtlinie

Bezug: A.8.24. Einsatz von Verschlüsselung und Verwaltung von Schlüsseln.

Festlegungen: zugelassene Verfahren/Schlüssellängen; Verschlüsselung ruhender und übertragener Daten; **Schlüssellebenszyklus** (Erzeugung, Verteilung, Aufbewahrung, Rotation, Vernichtung); Zuständigkeiten; Vorbereitung auf **Post-Quantum**-Migration langlebig vertraulicher Daten.

5.19 D-19 – Incident-Management-Verfahren

Bezug: A.5.24–A.5.28. Wie Sicherheitsvorfälle behandelt werden.

Ablauf: **Erkennen** → **Melden** → **Klassifizieren** → **Eindämmen** → **Beheben** → **Wiederherstellen** → **Auswerten** (Lessons Learned). Meldewege und Erreichbarkeiten, Eskalationsstufen, Beweissicherung, gesetzliche Meldepflichten (z. B. DSGVO 72 h, NIS-2). Verzahnung mit dem Service-Desk/Ticketing.

5.20 D-20 – Betriebskontinuität (Bezug zum BCM)

Bezug: A.5.29–A.5.30. Aufrechterhaltung der Informationssicherheit in Störungen.

Verweist auf das **BCM-Modul** (BSI 200-4 / ISO 22301): BIA, Notfallpläne, RTO/RPO, Wiederanlaufreihenfolge. Diese Richtlinie stellt sicher, dass Sicherheit auch im Notbetrieb gewahrt bleibt (z. B. Zugriffskontrolle bei Ausweichlösungen).

6 Startpaket – die Reihenfolge der Einführung

Wer ein ISMS neu aufbaut, geht in dieser Reihenfolge vor:

1. **Scope** (D-01) und **Leitlinie** (D-02) festlegen und freigeben.
2. **Rollen** (D-03) besetzen – ISB benennen.
3. **Risikoverfahren** (D-04/D-05) beschließen.
4. Schutzobjekte in Ordivis Platform modellieren, **Risiken bewerten** → Bericht D-10.
5. **SoA** (D-06) erstellen, **Behandlungsplan** (D-07) terminieren.
6. **Ziele** (D-08) setzen, **Awareness** (D-09) starten.
7. Betrieb: **Monitoring** (D-11), **Audit** (D-12), **Review** (D-13), **Korrekturen** (D-14).
8. Ergänzende Richtlinien (D-15 ff.) nach Bedarf und Risikolage.

> **Tipp.** Fangen Sie klein an: ein enger, sauberer Scope mit vollständiger Dokumentation ist mehr wert als ein breiter Scope mit Lücken. Erweitern lässt sich später jederzeit.

7 Verweise

- Kapitel **ID-28 ISMS** – Bedienung des Moduls Informationssicherheit.
- Kapitel **ID-29 BCM** – Betriebskontinuität und Notfallplanung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- Kapitel **ID-44 Rollen & Rechte** – Systemrolle ISB und Berechtigungen.
- ISO/IEC 27001:2022 und ISO/IEC 27002:2022 (Umsetzungsleitfaden Controls) – bei der jeweiligen Normungsstelle zu beziehen.

Verwandte Themen

- **ID-28 – Modul Informationssicherheit (ISMS)** – das Modul, mit dem diese Dokumente gepflegt werden
- **ID-29 – Modul Business Continuity Management** – Fortführungspläne als Nachweis zu A.5.29 f.
- **ID-32 – Modul Datenschutz (Art. 30 DSGVO)** – gemeinsame Nachweise mit dem Datenschutz
- **ID-44 – Rollen- & Rechte-Matrix** – Rechte für Lenkung und Freigabe

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · BCM

Modul Business Continuity Management (BCM)

Business Continuity Management nach BSI 200-4 und ISO 22301

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-29 BCM

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Business Continuity Management (BCM)

Was passiert, wenn das Rechenzentrum brennt, ein Verschlüsselungsangriff die Systeme lahmlegt oder für Wochen zu viele Mitarbeitende gleichzeitig ausfallen? Solche Ereignisse sind selten, aber wenn sie eintreten, entscheidet sich in Stunden, ob eine Organisation handlungsfähig bleibt. Wer dann erst anfängt zu überlegen, wer was tun soll, verliert wertvolle Zeit.

Business Continuity Management (BCM), auf Deutsch die Steuerung der Geschäftsfortführung, ist die Vorsorge für genau solche Fälle. Der Grundgedanke: erst herausfinden, welche Geschäftsprozesse wie schnell wieder laufen müssen (die Business-Impact-Analyse), dann Rollen, Pläne und Zuständigkeiten für den Notfall festlegen und das Ganze in Übungen erproben. Das Modul folgt den anerkannten Standards BSI 200-4 und ISO 22301. Es kümmert sich um die organisatorische Seite eines Notfalls; den technischen Wiederanlauf der Systeme behandelt die DR-Planung (ID-30).

INHALT

1 Zweck & Nutzen
2 Business-Impact-Analyse (BIA)
3 Notfallorganisation (BAO)
4 Notfallpläne
5 IT-Notfallhandbuch (generiert)
6 Übungen & CAPA
7 Abgrenzung
8 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Kritische Geschäftsprozesse und ihre Ausfalltoleranz kennen.
- Vorbereitung auf Notfälle (Rollen, Pläne, Übungen).
- Nachweise für Audits.

2 Business-Impact-Analyse (BIA)

Die **BIA** bewertet je Geschäftsprozess die Kritikalität und Zeitziele:

Abbildung 1 zeigt die Analyse mit Zeitzielen und der Prüfung auf Zeitketten-Konflikte.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

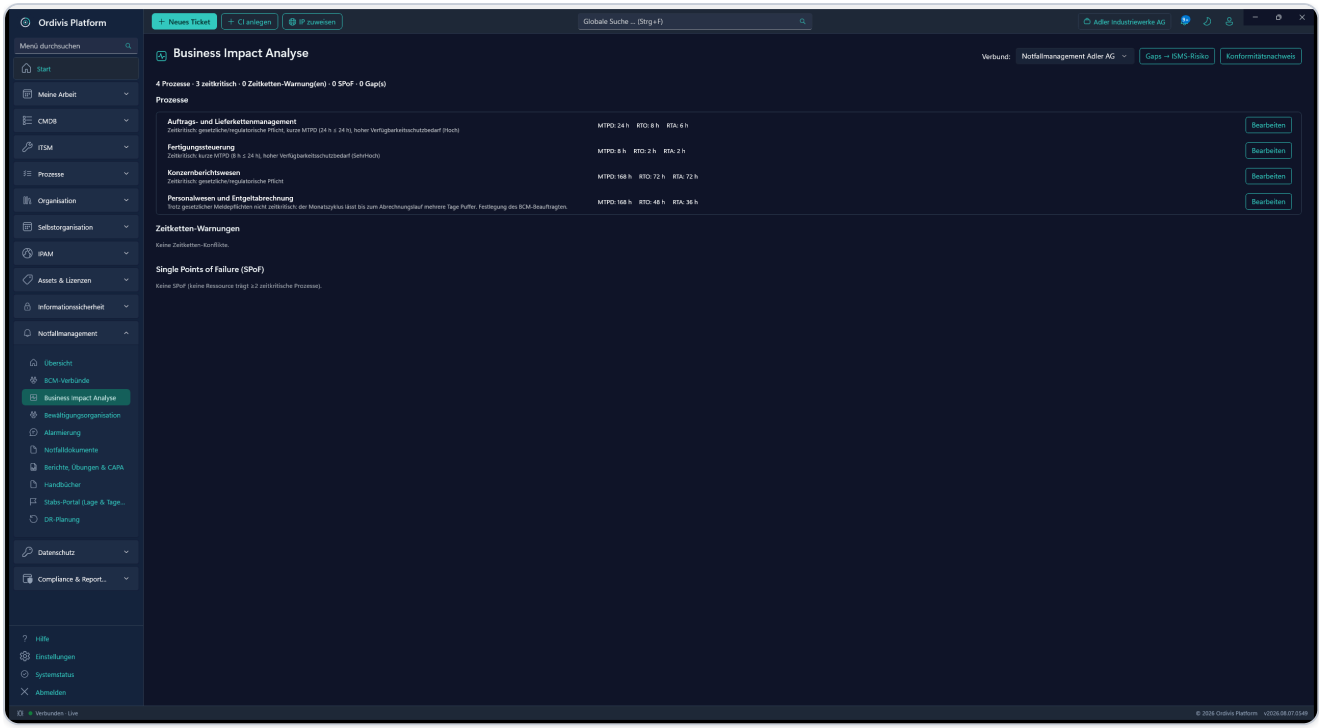

Abb. 1 – Business-Impact-Analyse: je Geschäftsprozess Zeitkritikalität, MTPD, RTO/RTA sowie Prüfung auf Zeitketten-Konflikte und Single Points of Failure.

- **MTPD** – maximal tolerierbare Ausfallzeit.
- **RTO/RPO** – Wiederanlauf- und Datenverlustziele.
- **Verfügbarkeitsanforderung**.

> **Wichtig:** Die **Zeitkritikalität** kommt aus dem **Vorfilter** (`BcmBiaVorfilter`) und wird **live** berechnet (Pflicht ODER MTPD ≤ 24 h ODER Verfügbarkeit hoch). Ein gespeicherter Wert zählt nur bei manueller Übersteuerung (`IsManualOverride`) – sonst weichen Berichte von der BIA-Seite ab.

3 Notfallorganisation (BAO)

> 💻 **Nur im Client:** *Notfallmanagement* ▸ *Bewältigungsorganisation* · 🔑 `bcm.plan.manage` — Das Portal zeigt die Aufstellung im *BCM-Arbeitsplatz*, besetzt die Rollen aber nicht.

Definieren Sie die **Besondere Aufbauorganisation** für den Notfall: Rollen (Krisenstab, Notfallbeauftragte), Erreichbarkeiten und Eskalationswege.

🔒 **Pflichtreihenfolge** — eine Rolle ohne Erreichbarkeit ist im Ernstfall keine. Besetzen Sie deshalb erst die Rolle und hinterlegen Sie unmittelbar danach, wie die Person nachts erreicht wird.

1. Öffnen Sie *Notfallmanagement* ▸ *Bewältigungsorganisation*.
2. Legen Sie die benötigten **Rollen** an (Krisenstabsleitung, Notfallbeauftragte, Sprecher …).
3. Besetzen Sie jede Rolle mit einer Person **und** einer Vertretung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4. Hinterlegen Sie je Person die **Erreichbarkeit außerhalb der Dienstzeit** – die dienstliche Durchwahl hilft um drei Uhr nachts nicht.
5. Legen Sie den **Eskalationsweg** fest: wer wen in welcher Reihenfolge alarmiert.
6. Speichern Sie.

✅ **Ergebnis:** Die Aufstellung steht und geht in das generierte IT-Notfallhandbuch ein (Abschnitt 5).

> **Achtung:** Prüfen Sie die Erreichbarkeiten mindestens jährlich. Eine BAO mit der Mobilnummer eines vor zwei Jahren ausgeschiedenen Kollegen ist im Ernstfall nicht bloß nutzlos – sie kostet die Minuten, in denen jemand vergeblich anruft.

Abbildung 2 zeigt eine solche Aufstellung.

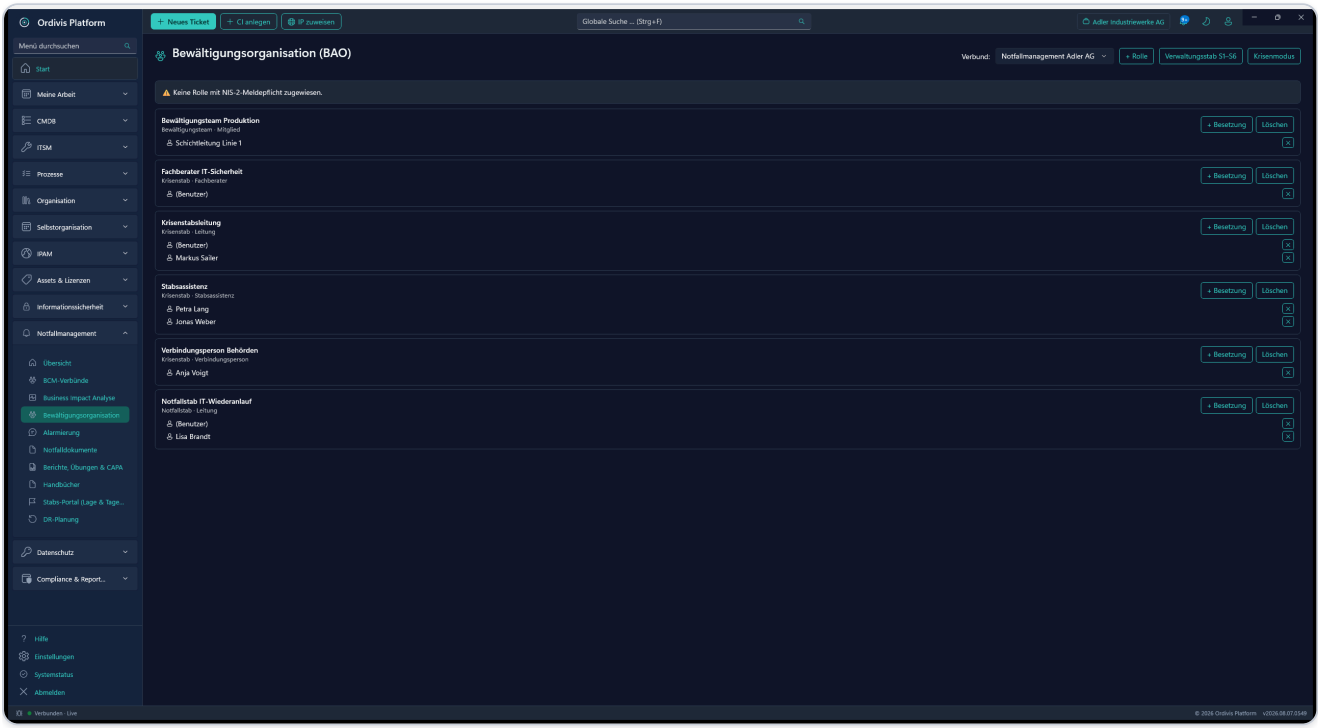

Abb. 2 – Bewältigungsorganisation (BAO): Krisenstab- und Notfallrollen mit den zugewiesenen Personen.

4 Notfallpläne

Erstellen Sie Pläne je Szenario (Ausfall Rechenzentrum, Personalausfall …) mit klaren Schritten, Verantwortlichen und benötigten Ressourcen. Pläne verknüpfen sich mit CIs/Diensten aus der CMDB.

5 IT-Notfallhandbuch (generiert)

Ordivis Platform erzeugt aus den Daten, die Sie in diesem Modul ohnehin pflegen, ein vollständiges **IT-Notfallhandbuch nach BSI 200-4** – Sie schreiben es nicht, Sie geben den Verbund an.

Erzeugen: Notfallhandbuch für einen BCM-Verbund generieren. Es entsteht ein versioniertes Handbuch im Entwurf mit sieben Kapiteln:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Kapitel	Woher der Inhalt kommt
Zweck & Geltungsbereich	Verbund, Reifegrad, Lage; zeitkritische Prozesse mit MTPD/RTO/RPO
Notfallorganisation (BAO)	Rollen und Besetzung inkl. Vertretung; Hinweis auf fehlende Vertretung und Meldepflicht
Alarmierung & Eskalation	Alarmierungskontakte je Eskalationsstufe (nur aktive)
Sofortmaßnahmen	Checklisten der Kategorie „Notfall“ aus der Checklisten-Engine
Kommunikationsplan	Kommunikations- und Warnkanäle des Verbunds
Kontaktverzeichnis	alle Kontakte inkl. Organisation (intern/Dienstleister)
Notfalldokumente	die **freigegebenen** Dokumente der Notfalldokumenten-Bibliothek

Der entscheidende Punkt – es wird nichts kopiert. Die Kapitel halten nur einen Verweis auf ihre Quelle und werden bei **jedem Abruf neu aufgelöst**. Ändern Sie eine Rufnummer in der Alarmierungsliste, steht sie sofort im Handbuch – ohne es neu zu erzeugen. Damit kann ein Notfallhandbuch nicht unbemerkt veralten, und es gibt keine doppelte Pflege.

> **Hinweis:** Lässt sich eine Quelle nicht mehr auflösen (z. B. gelöschter Verbund), weist das Kapitel dies **sichtbar** aus, statt still leer zu bleiben. Eine unbemerkte Lücke im Notfallhandbuch wäre gefährlicher als ein Hinweis.

Versionieren & freigeben: Ein Handbuch ist zunächst **Entwurf** und bearbeitbar. Mit der **Freigabe** wird der Stand eingefroren und gegen Änderungen geschützt – das ist der Stand, den Sie einer Prüfung vorlegen. Für eine Überarbeitung öffnen Sie den freigegebenen Stand wieder; dabei entsteht automatisch die **nächste Version**. Dynamische Kapitel zeigen auch im freigegebenen Handbuch weiterhin den Live-Stand ihrer Quellen – gerade Notfallkontakte müssen aktuell sein.

Betriebshandbücher und Wiederanlaufpläne nutzen dasselbe Handbuch-Gerüst; deren Generatoren folgen.

6 Übungen & CAPA

Planen und protokollieren Sie **Übungen** (Tests der Pläne). Aus Erkenntnissen entstehen **Korrektur- und Vorbeugemaßnahmen (CAPA)**, deren Umsetzung verfolgt wird.

7 Abgrenzung

- **BCM (dieses Modul):** organisatorische Geschäftsfortführung.
- **DR-Planung (ID-30):** technischer Wiederanlauf von IT-Systemen.
- **ISMS (ID-28):** Sicherheitsanforderungen und Risiken.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8 Rollen & Rechte (Auszug)

Recht	Erlaubt
`bcm.plan.read` / `bcm.report.read` / `bcm.risk.read`	Pläne, Berichte und Risiken ansehen
`bcm.bia.manage` / `bcm.plan.manage` / `bcm.exercise.manage`	BIA, Pläne und Übungen pflegen
`bcm.scope.manage` / `bcm.org.manage`	Geltungsbereich und Krisenorganisation verwalten
`bcm.crisis.operate`	Krisenmodus bedienen (Stab, Einsatztagebuch)
`handbook.read`	Handbücher lesen (Betrieb/Notfall/Wiederanlauf)
`handbook.manage`	Handbücher generieren und pflegen
`handbook.approve`	Handbuch freigeben (Entwurf → freigegebener Stand)

Verwandte Themen

- **ID-28 – Modul Informationssicherheit (ISMS)** – Informationssicherheit als Rahmen
- **ID-30 – Modul Notfall- & Wiederanlaufplanung (DR)** – der technische Wiederanlauf
- **ID-29a – Modul Checklisten** – Übungen und Prüfungen als Checkliste
- **ID-33 – Modul Organisation & Geschäftsverteilungsplan** – Prozesse und Federführung für die BIA

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · CHECKLISTEN

Modul Checklisten

Wiederkehrende Abläufe als Vorlage, Durchführung mit revisionssicherem Nachweis

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-29a Checklisten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Checklisten

Viele Abläufe in der IT wiederholen sich und dürfen dabei keinen Schritt auslassen: Ein neuer Mitarbeiter wird eingerichtet, ein Server geht in Betrieb, ein Change wird umgesetzt, im Notfall läuft ein System wieder an. Werden solche Abläufe aus dem Kopf abgearbeitet, wird früher oder später etwas vergessen – und im Nachhinein lässt sich nicht belegen, wer was wann erledigt hat.

Das Modul **Checklisten** trennt sauber zwischen der **Vorlage** (der wiederverwendbare Ablauf) und der **Durchführung** (eine konkrete, aktiv abgearbeitete Instanz). Jede Durchführung ist ein *Schnappschuss* ihrer Vorlage: spätere Änderungen an der Vorlage verändern laufende Durchführungen nicht. Das Modul ist querschnittlich nutzbar (Onboarding, Betrieb, Change, Wiederanlauf, Notfall) und arbeitet mandantengetrennt.

INHALT

1 Zweck & Nutzen
2 Eine Vorlage anlegen
3 Eine Durchführung starten
4 Eine Checkliste abarbeiten und abschließen
5 Den Nachweis abrufen
6 Automatische Auslösung
7 Vorbereitete Szenario-Vorlagen
8 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Wiederkehrende Abläufe standardisieren – kein Schritt wird vergessen.
- Revisionssicherer Nachweis: wer, wann, mit welcher Notiz erledigt hat.
- Selbst-Service für zugewiesene Checklisten im Portal.
- Nachweis je Durchführung als PDF (Audit/Compliance).

2 Eine Vorlage anlegen

💻 **Client:** *Meine Arbeit* ▸ *Playbooks/Checklisten* ▸ *Vorlagen* ▸ *Neue Vorlage* · 🌐 **Portal:** *Meine Vorgänge* ▸ *Checklisten-Vorlagen* ▸ *Neue Vorlage* · 🔑 `checklist.template.manage`

Eine Vorlage besteht aus **Abschnitten**, die wiederum **Prüfpunkte** enthalten. Je Prüfpunkt lassen sich Beschreibung, **Pflicht/optional**, Zuständigkeit (Rolle), erwartetes Ergebnis und Vorbedingung festlegen.

🔒 **Pflichtreihenfolge** — eine Vorlage lässt sich erst starten, wenn sie **aktiv** ist, und aktiv schalten lässt sie sich sinnvoll erst mit ihren Prüfpunkten. Schritt 6 gehört ans Ende, nicht an den Anfang.

1. Klicken Sie auf *Neue Vorlage*.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2. Tragen Sie einen *Namen* ein, der den Ablauf benennt (z. B. „On-Boarding neue Mitarbeiterin“).
3. Wählen Sie die *Kategorie*. Sie ist nicht nur Ordnung: An ihr hängt die automatische Auslösung (Abschnitt 6).
4. Beschreiben Sie unter *Beschreibung*, wofür die Vorlage gedacht ist und wann sie **nicht** passt.
5. Legen Sie die *Schritte* an – je Prüfpunkt eine Handlung, im Imperativ, und kennzeichnen Sie Pflichtpunkte.
6. Setzen Sie *Aktiv*, sobald die Vorlage vollständig ist.
7. Klicken Sie auf *Speichern*.

✅ **Ergebnis:** Es erscheint *Vorlage gespeichert*. Die Vorlage steht in der Liste mit *Name*, *Kategorie*, Anzahl *Schritte*, *Aktiv* und *Geändert* – und lässt sich starten.

Wichtig: Inhaltliche Änderungen erhöhen die **Version**. Laufende Durchführungen bleiben unberührt, weil jede ein Schnappschuss ist. Das ist gewollt: Wer eine Checkliste zur Hälfte abgearbeitet hat, soll nicht plötzlich vor anderen Punkten stehen.

Tipp: Formulieren Sie Prüfpunkte so, dass sie **eindeutig erledigt oder nicht erledigt** sind. „Konto einrichten“ lässt Spielraum, „Konto angelegt und Ersteinstieg getestet“ nicht.

3 Eine Durchführung starten

💻 **Nur im Client:** *Meine Arbeit* ▸ *Playbooks/Checklisten* ▸ *Durchführung starten* · 🔑 `checklist.run.manage` — Im Portal arbeitet man zugewiesene Durchführungen ab (Abschnitt 4), startet aber keine.

Voraussetzung: Die Vorlage ist **aktiv**. Inaktive Vorlagen stehen nicht zur Auswahl.

1. Öffnen Sie den Bereich *Vorlagen* und markieren Sie die gewünschte Vorlage.
2. Klicken Sie auf *Durchführung starten*.
3. Weisen Sie die Durchführung bei Bedarf einer **Person** zu.
4. Setzen Sie bei Bedarf eine **Fälligkeit**.
5. Bestätigen Sie.

✅ **Ergebnis:** Die Durchführung steht im Bereich *Durchführungen*. Ist sie zugewiesen, erscheint sie bei der betreffenden Person im Portal unter *Meine Vorgänge* ▸ *Checklisten*.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4 Eine Checkliste abarbeiten und abschließen

> 💻 **Client:** *Meine Arbeit* ▸ *Playbooks/Checklisten* ▸ *Durchführungen* · 🌐 **Portal:** *Meine Vorgänge* ▸ *Checklisten* · 🔑 `checklist.run.execute.self`

🔒 **Pflichtreihenfolge** — der Abschluss ist gesperrt, solange ein **Pflicht-Prüfpunkt** offen ist. Das Portal sagt das ausdrücklich: *Pflicht-Punkte (*) müssen erledigt oder n. a. sein.*

1. Öffnen Sie die Checkliste über *Öffnen*.
2. Arbeiten Sie die Prüfpunkte der Reihe nach ab und setzen Sie jeden auf *erledigt* oder *nicht zutreffend* (n. a.).
3. Halten Sie in der **Notiz** fest, was vom Erwarteten abwich. Genau diese Notiz ist es, die den Nachweis später brauchbar macht.
4. Tragen Sie unter *Unterschrift (Name)* Ihren Namen ein.
5. Klicken Sie auf *Abschließen*.

✅ **Ergebnis:** Es erscheint *Checkliste abgeschlossen.* Unter *Abgeschlossen von* stehen Name und Zeitstempel; über *PDF-Nachweis* laden Sie den Beleg herunter.

> **Hinweis:** Steht *Es sind noch offene Pflicht-Punkte vorhanden.*, fehlt mindestens ein mit `*` gekennzeichneter Punkt. Setzen Sie ihn auf *n. a.*, wenn er im Einzelfall wirklich nicht zutrifft – und schreiben Sie in die Notiz, warum.

> **Achtung:** Jede Statusänderung wandert **revisionssicher ins Audit-Log** (ID-27). Ein Haken, der nicht gesetzt wurde, fehlt dort ebenso nachweisbar wie einer, der gesetzt wurde. Haken Sie nichts ab, was Sie nicht getan haben.

5 Den Nachweis abrufen

> 💻 **Client:** *Meine Arbeit* ▸ *Playbooks/Checklisten* ▸ *Durchführungen* · 🌐 **Portal:** *Meine Vorgänge* ▸ *Checklisten* · 🔑 `checklist.read`

1. Öffnen Sie die abgeschlossene Durchführung.
2. Klicken Sie auf *PDF-Nachweis*.

✅ **Ergebnis:** Sie erhalten ein PDF mit Mandanten-Branding, allen Prüfpunkten samt Status und Notiz, Bearbeiter und Zeit sowie der Unterschrift. Es ist der Beleg, den ein Auditor sehen möchte.

6 Automatische Auslösung

Checklisten können durch Ereignisse aus anderen Modulen **automatisch** gestartet werden – die Zuordnung erfolgt über die **Kategorie** der Vorlage:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Krisen-Aktivierung** (BCM) startet die aktiven Vorlagen der Kategorie *Notfall* bzw. *Wiederanlauf*.
- **Change-Freigabe** startet die aktiven Vorlagen der Kategorie *Change*.

Die Auslösung ist idempotent (je Auslöser höchstens eine Durchführung) und vermerkt den Bezug zum auslösenden Vorgang, sodass die Durchführung rückverfolgbar bleibt.

So richten Sie eine automatische Auslösung ein:

> 💻 **Client:** *Meine Arbeit* ▸ *Playbooks/Checklisten* ▸ *Vorlagen* · 🌐 **Portal:** *Meine Vorgänge* ▸ *Checklisten-Vorlagen* · 🔑 `checklist.template.manage`

🔒 **Pflichtreihenfolge** — die Auslösung sucht **aktive** Vorlagen der passenden Kategorie. Eine Vorlage mit richtiger Kategorie, aber ohne *Aktiv*, wird nie gestartet – und meldet sich auch nicht.

1. Öffnen Sie die Vorlage.
2. Setzen Sie die *Kategorie* auf *Notfall*, *Wiederanlauf* oder *Change*.
3. Setzen Sie *Aktiv*.
4. Speichern Sie.

✅ **Ergebnis:** Beim nächsten passenden Ereignis entsteht die Durchführung von selbst – mit Verweis auf den auslösenden Vorgang.

> **Tipp:** Prüfen Sie das einmal im Trockenlauf, statt sich im Ernstfall darauf zu verlassen. Eine Krisen-Aktivierung ist der schlechteste Zeitpunkt, um zu bemerken, dass ein Haken fehlt.

7 Vorbereitete Szenario-Vorlagen

Zum schnellen Einstieg werden typische, anpassbare Vorlagen mitgeliefert: **On-Boarding**, **Off-Boarding**, **Server-Inbetriebnahme**, **Wiederanlaufplan**, **Change-Durchführung** und **Notfall**. Sie lassen sich je Mandant kopieren und an die eigenen Abläufe anpassen.

> 💻 **Client:** *Meine Arbeit* ▸ *Playbooks/Checklisten* ▸ *Vorlagen* · 🌐 **Portal:** *Meine Vorgänge* ▸ *Checklisten-Vorlagen* · 🔑 `checklist.template.manage`

💡 **Empfohlene Reihenfolge** — kopieren und anpassen, statt bei null anzufangen. Wer eine eigene Vorlage schreibt, vergisst die Punkte, an die man erst nach dem dritten Onboarding denkt.

1. Öffnen Sie die mitgelieferte Vorlage.
2. Legen Sie über *Neue Vorlage* eine Kopie mit eigenem Namen an.
3. Streichen Sie, was bei Ihnen nicht vorkommt, und ergänzen Sie Ihre Besonderheiten.
4. Setzen Sie *Aktiv* und speichern Sie.

✅ **Ergebnis:** Sie haben eine Vorlage, die zu Ihrem Haus passt, ohne den Ablauf neu erfunden zu haben.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8 Rollen & Rechte (Auszug)

Recht	Erlaubt
`checklist.read`	Vorlagen und Durchführungen ansehen, PDF-Nachweis abrufen
`checklist.run.execute.self`	Zugewiesene Checklisten abarbeiten und abschließen
`checklist.template.manage`	Vorlagen anlegen, bearbeiten, aktiv/inaktiv schalten
`checklist.run.manage`	Durchführungen starten und abbrechen

Verwandte Themen

- **ID-29 – Modul Business Continuity Management** – Übungen im Business Continuity Management
- **ID-30 – Modul Notfall- & Wiederanlaufplanung (DR)** – Prüfschritte im Wiederanlauf
- **ID-26 – Modul Workflows & Aufgaben** – Workflows für mehrstufige Abläufe mit Genehmigung
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – wiederkehrende Betriebsroutinen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · DR-PLANUNG

Modul Notfall- & Wiederanlaufplanung (DR)

Technischer Wiederanlauf von IT-Systemen nach einem Ausfall

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-30 DR-Planung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Notfall- & Wiederanlaufplanung (DR)

Wenn ein Server oder ein ganzes System ausgefallen ist, zählt jede Minute, und niemand sollte im Ernstfall erst überlegen müssen, in welcher Reihenfolge was wieder hochzufahren ist. Fehlt ein getesteter Plan, wird aus einer beherrschbaren Störung schnell ein langer Ausfall, weil Wissen fehlt, Abhängigkeiten übersehen werden oder eine Sicherung sich als unbrauchbar erweist.

Disaster Recovery (DR), also die Wiederherstellung nach einem Notfall, ist der technische Fahrplan für den Wiederanlauf der IT. Ein DR-Plan beschreibt je Ausfallszenario die konkreten Schritte, die richtige Reihenfolge, die Verantwortlichen und die betroffenen Systeme. Während das BCM (ID-29) die organisatorische Frage beantwortet „Wie bleibt das Geschäft handlungsfähig?“, beantwortet die DR-Planung die technische: „Wie bekommen wir diese Systeme wieder zum Laufen, und schnell genug?“.

INHALT

1 Zweck & Nutzen
2 Einen DR-Plan anlegen
3 Bezug zu CIs & Diensten
4 Zeitziele (RTO/RPO)
5 Test & Pflege
6 Rollen & Rechte (Auszug)
Verwandte Themen

1 Zweck & Nutzen

- Klare, getestete Schritte zur Wiederherstellung kritischer Systeme.
- Bezug zwischen Wiederanlaufplänen und konkreten CIs/Diensten.
- Einhaltung von RTO/RPO aus der BIA.

2 Einen DR-Plan anlegen

💻 **Nur im Client:** *Notfallmanagement* ▸ *DR-Planung* ▸ *Neuer Plan* · 🔑 `dr.manage` — Das Portal zeigt die Pläne unter *Notfallmanagement* ▸ *DR-Pläne* an und nimmt Testnachweise entgegen (Abschnitt 5), legt aber keine Pläne an.

Voraussetzung: Die betroffenen Systeme sind als **CI** in der CMDB erfasst (ID-21). Ohne CI lässt sich weder die Verknüpfung noch die abgeleitete Reihenfolge herstellen.

🔒 **Pflichtreihenfolge** — die vorgeschlagene Wiederanlaufreihenfolge (Abschnitt 3) entsteht aus den CMDB-Beziehungen der **verknüpften** Systeme. Verknüpfen Sie deshalb erst die CIs und ordnen Sie danach die Schritte, nicht umgekehrt.

1. Öffnen Sie *Notfallmanagement* ▸ *DR-Planung*.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2. Klicken Sie auf *Neuer Plan*.
3. Benennen Sie das **Szenario** so, wie es im Ernstfall gerufen wird – „Ausfall Datenbank-Server“, nicht „Plan 3“.
4. Verknüpfen Sie die betroffenen **CIs/Dienste** aus der CMDB.
5. Erfassen Sie die **Wiederanlaufschritte** – je Schritt Verantwortliche, veranschlagte Dauer und Voraussetzungen.
6. Tragen Sie **RTO** und **RPO** aus der BIA ein (Abschnitt 4).
7. Speichern Sie.

✅ **Ergebnis:** Der Plan steht in der Liste mit *Plan*, *Tier*, *RTO*, *RPO* und *Zuletzt getestet*. Solange kein Test dokumentiert ist, steht dort *nie getestet* – und das ist die ehrlichste Angabe, die die Liste machen kann.

> **Tipp:** Schreiben Sie die Schritte so, dass sie um drei Uhr nachts von jemandem ausgeführt werden können, der das System nicht gebaut hat. Alles, was „ist ja klar“ ist, ist es dann nicht.

3 Bezug zu CIs & Diensten

> 💻 **Client:** *Notfallmanagement* ▸ *DR-Planung*, Plan öffnen · 🌐 **Portal:** *Notfallmanagement* ▸ *DR-Pläne* · 🔑 `dr.read`

Durch die Verknüpfung mit der CMDB wissen Sie im Ernstfall genau, welche Systeme betroffen sind und in welcher Reihenfolge sie wiederhergestellt werden müssen (Abhängigkeiten zuerst).

Die vorgeschlagene Reihenfolge leitet Ordivis Platform aus den CMDB-Beziehungen der verknüpften Systeme ab: Was von wenig anderem abhängt, steht vorn. Lässt sich für ein System keine Abhängigkeit ermitteln – etwa weil das CI zwischenzeitlich entfernt wurde –, wird es **ans Ende gestellt und als ungeklärt gekennzeichnet**, nicht an den Anfang. Der Unterschied ist im Ernstfall entscheidend: „keine Abhängigkeiten bekannt" heißt nicht „hängt von nichts ab". Prüfen Sie solche Einträge, bevor Sie der Reihenfolge folgen.

So prüfen Sie die vorgeschlagene Reihenfolge:

🔒 **Pflichtreihenfolge** — sehen Sie zuerst die als *ungeklärt* gekennzeichneten Einträge durch. Erst danach hat die übrige Reihenfolge Aussagekraft.

1. Öffnen Sie den Plan.
2. Nehmen Sie sich jeden als **ungeklärt** gekennzeichneten Eintrag vor.
3. Prüfen Sie in der CMDB, ob das CI noch existiert und seine Beziehungen gepflegt sind (ID-21).
4. Ergänzen Sie fehlende Beziehungen.
5. Öffnen Sie den Plan erneut – die Reihenfolge wird neu abgeleitet.

✅ **Ergebnis:** Kein Eintrag steht mehr auf *ungeklärt*, und der Reihenfolge lässt sich im Notfall folgen, ohne sie zu hinterfragen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4 Zeitziele (RTO/RPO)

Übernehmen Sie **RTO** (Wiederanlaufzeit) und **RPO** (tolerierbarer Datenverlust) aus der BIA und prüfen Sie, ob die geplanten Schritte diese Ziele einhalten.

> 💻 **Nur im Client:** *Notfallmanagement* ▸ *DR-Planung*, Plan öffnen · 🔑 `dr.manage`

🔒 **Pflichtreihenfolge** — die BIA steht vor dem DR-Plan. RTO und RPO sind eine fachliche Festlegung des Geschäfts (ID-29), keine technische Schätzung. Wer sie im DR-Plan erfindet, dokumentiert seinen Wunsch, nicht die Anforderung.

1. Schlagen Sie RTO und RPO des betroffenen Prozesses in der BIA nach (ID-29).
2. Tragen Sie beide Werte im DR-Plan ein.
3. Addieren Sie die veranschlagten Dauern Ihrer Wiederanlaufschritte.
4. Vergleichen Sie die Summe mit dem RTO.

✅ **Ergebnis:** Entweder passt der Plan – oder Sie haben schwarz auf weiß eine Lücke, die Sie melden können, bevor sie im Ernstfall auffällt.

> **Achtung:** Ein RPO von vier Stunden verlangt eine Sicherung mindestens alle vier Stunden. Der DR-Plan ist der Ort, an dem diese Rechnung aufgeht oder nicht – prüfen Sie sie gegen den tatsächlichen Sicherungsplan (ID-35).

5 Test & Pflege

> 💻 **Client:** *Notfallmanagement* ▸ *DR-Planung*, Plan öffnen · 🌐 **Portal:** *Notfallmanagement* ▸ *DR-Pläne* ▸ *Test dokumentieren* · 🔑 `dr.manage`

Ein Plan ist nur so gut wie sein letzter Test. Nach jedem Test – ob Übung oder Ernstfall – gehört das Ergebnis in den Plan:

1. Öffnen Sie *DR-Pläne* und suchen Sie den geprüften Plan.
2. Klicken Sie auf *Test dokumentieren*.
3. Tragen Sie das *Testdatum* ein.
4. Wählen Sie unter *Ergebnis* zwischen *Erfolgreich* und *Mit Mängeln*.
5. Halten Sie unter *Feststellungen* fest, was nicht funktioniert hat. **Auch bei „Erfolgreich“** – dort steht, was länger gedauert hat als geplant.
6. Klicken Sie auf *Speichern*.

✅ **Ergebnis:** Es erscheint *Test dokumentiert*. In der Spalte *Zuletzt getestet* steht das Datum; *nie getestet* verschwindet.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Tipp: Prüfen Sie DR-Pläne regelmäßig – verzahnt mit den BCM-Übungen (ID-29) – und aktualisieren Sie sie bei jeder Änderung an der Infrastruktur. Ein Plan, der auf einen Server zeigt, den es nicht mehr gibt, ist im Ernstfall schlimmer als kein Plan: Er kostet die ersten zwanzig Minuten.

Achtung: *Mit Mängeln* ist ein Ergebnis, kein Makel. Ein Plan, bei dem nie etwas auffällt, wurde entweder nie ernsthaft getestet oder das Ergebnis nicht ehrlich vermerkt – beides fällt im Audit auf und im Ernstfall auf die Füße.

6 Rollen & Rechte (Auszug)

Recht	Erlaubt
`dr.read`	Pläne ansehen
`dr.manage`	Pläne und Testergebnisse pflegen

Verwandte Themen

- **ID-29 – Modul Business Continuity Management** – die fachliche Fortführungsplanung darüber
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Sicherung und Wiederherstellung in der Praxis
- **ID-21 – Modul CMDB** – Abhängigkeiten zwischen Systemen
- **ID-29a – Modul Checklisten** – Wiederanlauf als abhakbare Checkliste
- **ID-35 – Modul Datensicherung** – ob der RPO durch den Sicherungsplan gedeckt ist

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · BENACHRICHTIGUNGEN

Modul Benachrichtigungen & E-Mail-Integration

In-App-Glocke, E-Mail-Versand und -Eingang, Präferenzen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-31 Benachrichtigungen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Benachrichtigungen & E-Mail-Integration

Ein Ticketsystem lebt davon, dass die Beteiligten mitbekommen, wenn etwas passiert. Wird mir ein Ticket zugewiesen, wartet eine Genehmigung auf mich, hat der Melder geantwortet? Müsste man dafür ständig selbst nachsehen, ginge der halbe Nutzen verloren, und Wichtiges bliebe liegen.

Dieses Modul sorgt für den Informationsfluss in beide Richtungen. Nach außen meldet es Ereignisse aktiv – über die Glocke in der Anwendung und per E-Mail, wobei jeder selbst einstellen kann, worüber er informiert werden möchte. Nach innen nimmt es E-Mails entgegen: Antwortet ein Melder auf eine Benachrichtigung, ordnet Ordivis Platform die Nachricht automatisch dem richtigen Ticket zu. So bleibt die Kommunikation an einem Ort, statt sich über verstreute Postfächer zu verteilen.

INHALT

1 In-App-Benachrichtigungen (Glocke)
2 E-Mail-Versand (SMTP)
2.1 E-Mail-Signaturen
3 E-Mail-Eingang (IMAP)
3.1 Was mit der Nachricht danach geschieht
4 Sichere E-Mail (OAuth2 & TLS)
5 Benachrichtigungs-Präferenzen
6 Kontakte
8 Messenger (Mattermost, generischer Webhook)
8.1 Verbindung einrichten
8.2 Zwei Ziele, die nie vertauscht werden
8.3 Versand, Wiederholung und Grenzen
8.4 Was der Zustand sagt — und was nicht
8.5 Abgrenzung zur Alarmierung
7 Rollen & Rechte (Auszug)
Verwandte Themen

1 In-App-Benachrichtigungen (Glocke)

Die **Glocke** in der Kopfzeile zeigt neue Ereignisse (zugewiesenes Ticket, Genehmigung fällig, Antwort des Melders). Über REST-Endpunkte (`GET` / `PATCH /api/v1/notifications`) werden sie geladen und als gelesen markiert.

2 E-Mail-Versand (SMTP)

💻 **Nur im Client:** *Fußbereich ▸ Einstellungen ▸ E-Mail & Benachrichtigungen* · 🔑 `system.config.manage` — Das Portal hat keine SMTP-Einrichtung; der Versand gilt für beide Oberflächen gleichermaßen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Benachrichtigungen und Ticket-Antworten gehen per **SMTP** hinaus. Ein interner Exchange wird über **STARTTLS** (Port 587) mit FQDN und interner Root-CA angebunden; eine **Outbox** mit **Backoff** und **Rate-Limit** sorgt für zuverlässigen Versand.

> **Voraussetzung:** Die interne Root-CA muss im Vertrauensspeicher des Servers liegen; der Mail-Server ist ggf. nur über VPN erreichbar.

🔒 **Pflichtreihenfolge** — der Testversand (Schritt 5) ist der einzige Beleg, dass die Angaben stimmen. Ohne ihn merken Sie den Fehler erst, wenn die erste Benachrichtigung ausbleibt – und dann merkt es niemand.

1. Öffnen Sie *Einstellungen* ▸ *E-Mail & Benachrichtigungen*.
2. Tragen Sie **Server** und **Port** ein (interner Exchange in der Regel Port 587).
3. Schalten Sie **STARTTLS** ein.
4. Hinterlegen Sie Absenderadresse und Zugangsdaten.
5. Lösen Sie den **Testversand** aus und prüfen Sie den Eingang.

✅ **Ergebnis:** Die Testmail kommt an. Erst dann ist der Versand eingerichtet – ein gespeichertes Formular ist kein Nachweis.

> **Achtung:** Ein Zertifikatsfehler äußert sich nicht als Fehlermeldung beim Speichern, sondern als stehengebliebene Outbox. Prüfen Sie bei ausbleibenden Mails zuerst, ob die interne Root-CA auf dem **Server** vertraut wird – nicht auf Ihrem Arbeitsplatz.

2.1 E-Mail-Signaturen

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *E-Mail-Signatur* · 🔑 `ticketing.config.manage` — Das Portal führt Textbausteine und Vorlagen, aber keine Signaturen.

Ausgehende Ticket-Mails werden zentral mit einer **Signatur** versehen – einheitlich für alle Bearbeiter und rechtssicher.

So hinterlegen Sie eine Signatur:

🔒 **Pflichtreihenfolge** — die **Firmen-Rechtsdaten (Impressum)** liefern die Platzhalter `{{company.name}}`, `{{company.address}}` und `{{company.legal_line}}`. Sind sie leer, bleiben sie in der Vorschau leer, und die Signatur sieht fertig aus, obwohl das Impressum fehlt.

1. Pflegen Sie zuerst die **Firmen-Rechtsdaten (Impressum)** — sie stehen auf derselben Seite unterhalb der Signatur-Vorlage.
2. Öffnen Sie *Fußbereich* ▸ *Einstellungen* ▸ *E-Mail-Signatur*.
3. Schreiben Sie die **globale Standard-Signatur** mit den benötigten Platzhaltern.
4. Hinterlegen Sie bei Bedarf je Support-Kanal eine abweichende Vorlage.
5. Prüfen Sie das Ergebnis in der **Live-Vorschau** – je Kanal einzeln.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6. Speichern Sie.

✅ **Ergebnis:** Jede ausgehende Ticket-Mail trägt die Signatur samt vollständigem Impressum.

- **Global oder je Kanal:** Eine globale Standard-Signatur gilt für alle Support-Gruppen; je Support-Kanal lässt sich abweichend eine eigene Vorlage hinterlegen.
- **Dynamische Platzhalter:** Vorlagen enthalten Felder, die beim Versand automatisch gefüllt werden – `{{agent.first_name}}`, `{{agent.last_name}}` (aus Datenschutzgründen abgekürzt), `{{agent.title}}`, `{{ticket.id}}` sowie die Firmendaten `{{company.name}}`, `{{company.address}}` und `{{company.legal_line}}`. Unbekannte Platzhalter werden entfernt; fehlt ein Vor-/Nachname, greift der Anzeigename.
- **Impressum an jeder Mail:** Ein rechtlich vollständiges Impressum (Firmen-Rechtsdaten) wird an jede ausgehende Mail angehängt. Die Firmendaten pflegen Sie im Abschnitt **Firmen-Rechtsdaten (Impressum)** derselben Seite.
- **Verwaltung & Vorschau:** unter **Fußbereich ▸ Einstellungen ▸ E-Mail-Signatur** – mit Live-Vorschau je Kanal.
- **Best Practice:** keine eingebetteten Bilder, keine persönlichen Kontaktdaten des Bearbeiters; der Nachname wird zum Schutz der Beschäftigten gekürzt.

> **Berechtigung:** Signaturen und Firmen-Rechtsdaten pflegt, wer `ticketing.config.manage` besitzt.

3 E-Mail-Eingang (IMAP)

Eingehende E-Mails werden automatisch Tickets zugeordnet:

- **Betreff-/Thread-Erkennung** ordnet Antworten dem richtigen Ticket zu.
- Unbekannte Absender können automatisch als **Kontakt** angelegt werden.
- **Schutzmechanismen:** Schleifen-, Bounce- und Anhangschutz verhindern Fehlverhalten.

3.1 Was mit der Nachricht danach geschieht

Verarbeitete Nachrichten werden aus dem Postfach gelöscht. Der Inhalt liegt ab diesem Zeitpunkt am Vorgang: Betreff, Text, Absender und Anhänge – und zusätzlich die **vollständige Original-Mail mit allen Kopfzeilen** als interner Anhang. Die Nachricht im Postfach ist damit eine Zweitschrift, und ein Eingang, der sie behält, sammelt auf Dauer personenbezogene Post ohne Zweck.

Wollen Sie die Zweitschriften behalten, tragen Sie in den Postfach-Einstellungen einen **Ablage-Ordner** ein – dann werden sie dorthin verschoben statt gelöscht.

Gelöscht wird ausschließlich, was nachweislich zu einem Vorgang geworden ist:

Fall	Was geschieht
Vorgang angelegt oder Antwort zugeordnet	Original hängt am Vorgang → Nachricht wird gelöscht (oder verschoben)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Fall	Was geschieht
Verarbeitung gescheitert	Nachricht geht in den **Fehlerordner** und bleibt dort auffindbar
Von einem Fachmodul übernommen (z. B. Berichtsmail der Datensicherung)	bleibt als gelesen im Eingang – es entsteht ja kein Vorgang, an dem das Original hinge

Der Altbestand bleibt liegen. Nachrichten, die vor dieser Fassung verarbeitet wurden, sind als gelesen markiert; der Abruf sieht sich nur **ungelesene** an und fasst sie deshalb nie wieder an. Räumen Sie diese einmal von Hand ab. Automatisch nachzuräumen wäre riskant: Bei einer gelesenen Nachricht lässt sich nicht mehr unterscheiden, ob das System sie verarbeitet oder ob ein Mensch sie nur geöffnet hat.

4 Sichere E-Mail (OAuth2 & TLS)

Für moderne Postfächer unterstützt Ordivis Platform **OAuth2/Device-Code**. Zusätzlich gibt es **TLS-Pinning** und **Schlüsselrotation** für den Schutz gespeicherter Zugangsdaten.

Wichtig: In lokalen Testumgebungen kann der Vault-Schlüssel von den gespeicherten Geheimnissen abweichen – dann müssen die E-Mail-Zugangsdaten neu hinterlegt werden.

5 Benachrichtigungs-Präferenzen

Benutzer steuern über **Präferenzen**, welche Ereignisse sie per In-App und/oder E-Mail erhalten (z. B. `system.update.available`). Bearbeiter werden bei relevanter Ticket-Aktivität informiert.

6 Kontakte

Ein lokales **Kontaktverzeichnis** (E-Mail-Verzeichnis) speist Absender-/Empfänger-Auflösung. Bei unbekanntem Absender erfolgt – abhängig von `contacts.manage` – eine automatische Anlage.

8 Messenger (Mattermost, generischer Webhook)

Alles, was In-App oder per E-Mail gemeldet wird, kann **zusätzlich** über einen Messenger gehen. Der Messenger ist dabei ein **dritter Kanal in derselben Schaltmatrix** (§5) und kein eigener Meldungsstrom: Der Schalter steht je Ereignis und je Person neben In-App und E-Mail.

⚠️ **Der Kanal ist ausnahmslos Opt-in.** Richtet ein Mandant eine Verbindung ein, wird noch nichts verschickt — jede Person entscheidet selbst, welche Ereignisse sie im Messenger erreichen sollen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8.1 Verbindung einrichten

Einstellungen › E-Mail › Messenger-Verbindungen (Windows-Client) bzw. *Messenger-Verbindungen* (Portal). Erforderlich ist das Recht `notifications.messenger.admin`; es ist **bewusst nicht** im Postfachrecht enthalten — ein Bot-Token öffnet einen anderen Dienst als ein Postfach-Kennwort.

Anbieter	Adresse	Geheimnis	Kann
Mattermost	Serveradresse, z. B. `https://chat.example.org`	Zugangstoken eines **Bot-Kontos**	Direktnachricht **und** Kanal
Generischer Webhook	vollständige Ziel-URL	optionales Geheimnis (als Bearer-Token)	**nur** Kanal

Das Bot-Konto muss Mitglied der Kanäle sein, in die es schreiben soll. Für Direktnachrichten genügt das Bot-Konto selbst.

Das Geheimnis wird **verschlüsselt** abgelegt (AES-256-GCM, dasselbe Verfahren wie beim Postfach-Kennwort) und von der Schnittstelle nie im Klartext zurückgegeben — sichtbar ist nur, *ob* eines hinterlegt ist.

8.2 Zwei Ziele, die nie vertauscht werden

- **Direktnachricht** an die Person: alles Persönliche — zugewiesen, Genehmigung, Erinnerung.
- **Kanal/Raum**: Team-Ereignisse ohne persönlichen Bezug, etwa ein neues Ticket ohne Zuweisung.

Die Zuordnung *Ordivis-Benutzer → Messenger-Konto* läuft über die **E-Mail-Adresse**. Ist eine Person im Messenger nicht auffindbar, steht das sichtbar in der Verwaltung unter *Zuordnung der Benutzer* — und nicht nur im Protokoll: Sonst glaubt jemand, benachrichtigt zu werden, und wird es nie.

⚠️ Der generische Webhook kann **keine** Direktnachricht. Er meldet das, und die Oberfläche bietet sie für ihn gar nicht erst an. Persönliche Benachrichtigungen brauchen deshalb eine Anbindung mit Direktnachricht.

8.3 Versand, Wiederholung und Grenzen

Nichts wird im Anfragepfad gesendet: Nachrichten werden **eingereiht** und von einem Job zugestellt. Ein vorübergehender Fehler (Netz, 5xx, Ratengrenze) wird mit wachsendem Abstand wiederholt (1, 2, 4, 8, 16, 32 Minuten); eine **dauerhafte** Ablehnung — archivierter Kanal, gelöschtes Konto, ungültiges Token — wird **nicht** wiederholt. Nach 20 Fehlversuchen in Folge legt sich die Verbindung still, damit ein gesperrtes Bot-Konto nicht im Minutentakt angerufen wird.

Der Sende-Deckel *Nachrichten je Stunde* schützt nicht vor dem eigenen Versand, sondern vor der Ratengrenze des Dienstes. Wird er erreicht, wird die Nachricht **zurückgestellt** und nicht als Fehlversuch gezählt.

8.4 Was der Zustand sagt — und was nicht

Im Lagebild *Letzte Nachrichten* steht je Nachricht **angenommen**, **abgelehnt**, **wartet** oder **unbekannt**.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

⚠️ **„Angenommen" ist nicht „zugestellt".** Weder Mattermost noch ein Webhook liefert eine Empfangsbestätigung; die stärkste belegbare Aussage lautet, dass der Dienst die Nachricht entgegengenommen hat. „Unbekannt" bleibt deshalb ein eigener Zustand und wird nicht zu einem grünen Haken geschönt.

⚠️ Ein bestandener **Verbindungstest** belegt nur die Erreichbarkeit. Er räumt einen Versandfehler nicht weg — und beim generischen Webhook erzeugt er sichtbar eine Nachricht im Zielkanal, weil ein eingehender Webhook keinen Prüfweg ohne Nebenwirkung hat.

8.5 Abgrenzung zur Alarmierung

Der Messenger ist ein **Benachrichtigungsweg**, kein Alarmierungsweg. Läuft er — wie üblich — im selben Netz wie die Anwendung, stirbt er mit ihr; als Alarmweg wäre er damit auf derselben Stufe wie E-Mail. Eskalationsstufen, Quittungspflicht und Zustellnachweis im BCM-Sinn gehören deshalb nicht hierher.

7 Rollen & Rechte (Auszug)

Recht	Erlaubt
(kein eigenes Recht)	Die eigenen Benachrichtigungen sieht jede angemeldete Person; ein Recht wird dafür nicht geprüft.
`contacts.manage`	Kontakte verwalten
`system.config.manage`	E-Mail-Integration konfigurieren
`notifications.messenger.admin`	Messenger-Verbindungen verwalten (MSN-03)

Verwandte Themen

- **ID-40 – Administrationshandbuch** – SMTP und Postfach einrichten
- **ID-20 – Modul Service Desk / Ticketing** – welche Ticketereignisse Nachrichten auslösen
- **ID-41 – Modul Identität, Benutzer & Rollen** – wer als Empfänger infrage kommt
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Zustellprobleme im Betrieb eingrenzen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · DATENSCHUTZ

Modul Datenschutz (Verarbeitungsverzeichnis)

Verzeichnis von Verarbeitungstätigkeiten (Art. 30 DSGVO), Meldewesen und behördenkonformer Export

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-32 Datenschutz

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Datenschutz (Verarbeitungsverzeichnis)

Wer personenbezogene Daten verarbeitet, muss darüber ein **Verzeichnis von Verarbeitungstätigkeiten** führen – so verlangt es Artikel 30 der Datenschutz-Grundverordnung. Die Aufsichtsbehörde kann es jederzeit anfordern. In der Praxis liegt dieses Verzeichnis oft in einer Tabelle: Niemand weiß, wer sie zuletzt geändert hat, die Fachbereiche melden nichts nach, und in der Prüfung fehlt die Hälfte der Pflichtangaben.

Das Modul **Datenschutz** führt dieses Verzeichnis im Produkt – mit drei Eigenschaften, die eine Tabelle nicht hat: Jede Änderung ist **nachweisbar**, **Lücken werden sichtbar** statt erst in der Prüfung, und die Fachbereiche können ihre Verarbeitungen **selbst melden**, ohne Datenschutz-Vorwissen.

> **Abgrenzung:** Nicht zu verwechseln mit der Selbstauskunft *„Was verarbeitet Ordivis Platform selbst?"* – die steht unverändert unter *Datenschutz → Verarbeitungsverzeichnis (Produkt)* und beschreibt die Software. Dieses Modul führt **Ihr eigenes** Verzeichnis über **Ihre** Verarbeitungen.

INHALT

1 Zweck & Nutzen
2 Rollen und Rechte
3 Die DSB-Zentrale (Client)
4 Ein Verfahren bearbeiten (360°-Profil)
4.4 Empfänger & Drittland
4.5 Verarbeitende Systeme (Register „Systeme")
4.6 Datenschutz-Folgenabschätzung (Register „DSFA")
4.7 Mehrstufige Freigabe (Register „Freigabe")
4.8 Technisch-organisatorische Maßnahmen (Register „TOM")
Lebenszyklus
5 Kataloge (Betroffene und Datenkategorien)
6 Meldewesen: Datenschutzverletzungen (Art. 33/34)
7 Behördenkonformer Export (Art. 30 Abs. 4)
8 Fachbereiche melden selbst (Portal)
9 Grenzen (Stand dieser Fassung)
Verwandte Themen

1 Zweck & Nutzen

- Verzeichnis nach Art. 30 DSGVO im Produkt führen, statt in einer Tabelle.
- **Revisionssicherer Nachweis** je Verfahren: wer, wann, welches Feld, von welchem Wert auf welchen.
- **Unvollständige Verfahren werden angezeigt** – nicht erst von der Aufsichtsbehörde gefunden.
- Meldewesen für Datenschutzverletzungen mit **72-Stunden-Frist** (Art. 33/34).
- **Behördenkonformer Export** auf Knopfdruck (PDF, CSV, JSON).
- Dezentrale Meldung durch Fachbereiche über das Portal.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2 Rollen und Rechte

Rolle	Recht	Sieht / darf
Fachbereich	`datenschutz.vvt.read` + `datenschutz.vvt.edit`	**nur die eigenen** Verfahren; anlegen, ändern, zur Prüfung geben
Betriebsrat / Leserecht	`datenschutz.vvt.read` + `datenschutz.vvt.read.all`	das **gesamte** Verzeichnis, nur lesend
Datenschutzbeauftragte(r)	zusätzlich `datenschutz.vvt.approve`, `datenschutz.catalog.manage`, `datenschutz.vvt.export`, `datenschutz.breach.*`	alles: freigeben, Kataloge pflegen, exportieren, Verletzungen bearbeiten

Die Einschränkung auf „eigene Verfahren" wirkt **serverseitig**: Ohne `datenschutz.vvt.read.all` sind fremde Verfahren weder in der Liste noch über einen Direktaufruf erreichbar. Wer Verfahren erfasst, kann sie **nicht selbst freigeben** – dafür braucht es das eigene Recht `datenschutz.vvt.approve` (Vier-Augen-Prinzip).

> Das Modul lässt sich vom Plattform-Administrator ein- und ausschalten (*Modul-Schalter*: `datenschutz`). Ist es aus, erscheint der Menüpunkt nicht.

3 Die DSB-Zentrale (Client)

Datenschutz → Verarbeitungsverzeichnis zeigt alle Verfahren mit Zweck, Fachbereich, Status und der nächsten Wiedervorlage – filter-, sortier- und exportierbar wie alle Tabellen im Client.

Zwei Dinge hebt die Liste bewusst hervor:

- **Spalte „Befunde"** – was einem Verfahren zu Art. 30 Abs. 1 fehlt: keine Rechtsgrundlage, keine Betroffenen-/Datenkategorien, keine Empfängerkategorien, keine Aufbewahrungsfrist, oder ein Empfänger im Drittland ohne belegte Garantie. Der Filter **„Nur mit Befunden"** macht daraus die Arbeitsliste. Die Banner oben zählen über den **gesamten** Bestand – sie verschwinden nicht, wenn Sie filtern.
- **Wiedervorlage** – ein freigegebenes Verfahren wird nach dem eingestellten Turnus fällig. Das Verzeichnis soll den *aktuellen* Stand abbilden, nicht den von vor drei Jahren.

4 Ein Verfahren bearbeiten (360°-Profil)

Ein Doppelklick öffnet das Verfahren als **360°-Profil** mit den Registern **Stammdaten**, **DSGVO-Metadaten**, **Systeme**, **DSFA**, **Freigabe** und **Nachweis**:

Stammdaten – Bezeichnung, Zweck (beides Pflicht nach Art. 30 Abs. 1 lit. b), Beschreibung, Fachbereich, Zielgruppenprofil (KMU / Enterprise / Öffentliche Verwaltung) und der Überprüfungsturnus. Es gibt **genau**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

eine Schaltfläche *Speichern*, und sie schreibt ausschließlich diese Stammdaten.

Das **Zielgruppenprofil** steuert dabei mehr als eine Bezeichnung: Es legt fest, wie viele **Freigabe-Stufen** ein Verfahren durchläuft (Abschnitt 4.7), welche Spezialgesetze angeboten werden und ob der Betriebsrat mitbestimmt. Ein Panel unter dem Feld zeigt an, was das gewählte Profil konkret vorsieht.

DSGVO-Metadaten – die Pflichtangaben:

- **Rechtsgrundlagen (Art. 6 Abs. 1)** – die Buchstaben stehen im Klartext dabei (lit. a Einwilligung, lit. b Vertrag, …), damit die Zuordnung ohne Übersetzungstabelle prüfbar ist. Optional ergänzt um ein Spezialgesetz (BDSG, LDSG, Betriebs- oder Konzernvereinbarung) mit Fundstelle. Bei **berechtigtem Interesse (lit. f)** verlangt das Modul die **Interessenabwägung** – ohne sie ist die Grundlage nicht belastbar, und das Verfahren wird als Befund ausgewiesen.
- **Betroffene & Datenkategorien (Art. 30 Abs. 1 lit. c)** – hier wird verknüpft, *welche* Daten über *welche* Personengruppe verarbeitet werden. Diese Verbindung ist die eigentliche Aussage: „Wir verarbeiten Gesundheitsdaten" und „wir verarbeiten Daten von Bewerbern" ergibt zusammen noch nicht „Gesundheitsdaten **von Bewerbern**".
- **Aufbewahrung & Löschung (Art. 30 Abs. 1 lit. f)** – eine Frist besteht aus **Dauer + Auslöser**, nicht aus einem Datum: „6 Jahre" ist ohne „ab wann" wertlos. Wählbar sind ab Erhebung, ab Vertragsende, ab Jahresende, ab Zweckfortfall oder ab Widerruf; danach löschen, anonymisieren oder sperren. Fristen lassen sich je Datenkategorie festlegen – Bewerbungsunterlagen sechs Monate, die Buchungsbelege dazu zehn Jahre.
- **Empfänger & Drittland (Art. 30 Abs. 1 lit. d/e)** – siehe Abschnitt 4.4.

4.4 Empfänger & Drittland

> 💻 **Client:** *Datenschutz* ▸ *Verarbeitungsverzeichnis*, Verfahren öffnen · 🌐 **Portal:** *Mein Arbeitsplatz* ▸ *Datenschutz-Zentrale* · 🔑 `datenschutz.vvt.edit`

🔒 **Pflichtreihenfolge** — die **Rolle** des Empfängers entscheidet über die Pflichten, das **Sitzland** über die Garantie. Tragen Sie beide ein, bevor Sie die Fundstelle suchen: Bei einem Empfänger in der EU/im EWR brauchen Sie gar keine.

1. Öffnen Sie das Verfahren und wechseln Sie zu den Empfängern.
2. Tragen Sie die **Kategorie** ein (z. B. „Steuerberater“). Sie ist Pflicht.
3. Ergänzen Sie bei Bedarf den **Namen** des Empfängers – die Aufsicht fragt gern danach.
4. Wählen Sie die **Rolle**: *Auftragsverarbeiter* (Art. 28) oder *eigenverantwortlicher Dritter*.
5. Tragen Sie das **Sitzland** ein.
6. Liegt der Empfänger im **Drittland**, hinterlegen Sie die **Garantie** samt Fundstelle.

✅ **Ergebnis:** Der Befund *„Keine Empfängerkategorien erfasst"* verschwindet, und der behördenkonforme Export (Abschnitt 7) weist die Übermittlung vollständig aus.

Artikel 30 verlangt die **Kategorien von Empfängern** – nicht zwingend die Namen. „Steuerberater" genügt; „Musterkanzlei GmbH" dürfen Sie zusätzlich hinterlegen und die Aufsicht fragt gern danach. **Die Kategorie**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ist Pflicht. Auch „Wir geben nichts weiter" ist eine Angabe – ein leeres Feld ist dagegen keine, und das Verfahren trägt dann den Befund *„Keine Empfängerkategorien erfasst"*.

Zu jedem Empfänger gehört die **Rolle**, denn sie entscheidet über die Pflichten: Ein *Auftragsverarbeiter* (Art. 28) handelt weisungsgebunden und braucht einen Vertrag; ein *eigenverantwortlicher Dritter* – etwa Ihr Steuerberater – braucht eine **eigene** Rechtsgrundlage. Eine Schwesterfirma ist datenschutzrechtlich ebenfalls ein Dritter: Ein Konzernprivileg kennt die DSGVO nicht.

Sitzland und Garantie. Ordivis Platform entscheidet automatisch nur eine Frage: EU/EWR oder Drittland. Das folgt aus den Verträgen und ändert sich nicht. Liegt der Empfänger außerhalb, verlangt Kapitel V eine Garantie – Angemessenheitsbeschluss (Art. 45), Standardvertragsklauseln (Art. 46), Binding Corporate Rules (Art. 47) oder eine Ausnahme im Einzelfall (Art. 49). Fehlt sie, ist die Übermittlung unzulässig, und das Verfahren wird als Befund ausgewiesen. Tragen Sie die **Fundstelle** ein (SCC-Modul mit Datum, Zertifikatsnummer) – eine Garantie ohne Beleg ist eine Behauptung.

> **Vereinigte Staaten:** „Angemessenheitsbeschluss" allein reicht nicht. Der Beschluss gilt nicht dem Land, sondern nur Empfängern, die unter dem **Data Privacy Framework zertifiziert** sind. Ordivis Platform verlangt deshalb den Zertifizierungsnachweis; ohne ihn brauchen Sie Standardvertragsklauseln.

Warum Ordivis Ihnen nicht widerspricht: Angemessenheitsbeschlüsse werden erlassen – und wieder gekippt (das Privacy Shield fiel 2020). Eine im Programm hinterlegte Länderliste ist ab dem Tag ihrer Auslieferung möglicherweise veraltet, und eine veraltete Liste, die stillschweigend „zulässig" sagt, wäre gefährlicher als gar keine. Geben Sie einen Angemessenheitsbeschluss für ein Land an, das unser Stand nicht kennt, erhalten Sie deshalb einen **Hinweis mit Datum** – keinen Widerspruch. Die Entscheidung bleibt bei Ihnen.

4.5 Verarbeitende Systeme (Register „Systeme")

Welche IT-Systeme verarbeiten in diesem Verfahren personenbezogene Daten? Das Register **Systeme** verknüpft das Verfahren mit den zugehörigen **CI aus der CMDB** – der Name wird stets aktuell aufgelöst, nie abgeschrieben. Zu jedem System zeigt Ordivis Platform den **Schutzbedarf aus dem Informationssicherheits-Modul** (read-only), sodass Datenschutz und Informationssicherheit dieselbe Wahrheit sehen.

Werden **besonders schutzwürdige Daten** (Art. 9/10) in einem Verfahren auf einem System verarbeitet, für das **kein hoher Schutzbedarf** festgestellt ist, warnt das Register vor dem **„ungeschützten Datenstrom"**. Das ist kein Fehler des Verfahrens, sondern ein Prüfhinweis: Entweder ist der Schutzbedarf im ISMS zu niedrig angesetzt, oder das System ist für diese Daten nicht geeignet.

4.6 Datenschutz-Folgenabschätzung (Register „DSFA")

Artikel 35 verlangt eine **Datenschutz-Folgenabschätzung**, wenn eine Verarbeitung *voraussichtlich ein hohes Risiko* birgt. Das Register **DSFA** führt die **Schwellenwertanalyse** anhand der Muss-Kriterien (Art. 35 Abs. 3 und der Liste der Aufsichtsbehörden – Profiling, umfangreiche besondere Kategorien, systematische Überwachung, neue Technologien, Datenabgleich, schutzbedürftige Betroffene) und lässt Sie das **Restrisiko** aus Eintrittswahrscheinlichkeit × Schadensausmaß (je 1–3) einschätzen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Das Ergebnis ist eine **Ampel**: „nicht bewertet" (grau), grün, gelb (Prüfung ratsam) oder rot (DSFA-Pflicht). **Ein nicht bewertetes Verfahren bleibt bewusst grau** – die Ampel darf nie fälschlich „unbedenklich" suggerieren, nur weil noch niemand hingesehen hat. Die DSFA-Pflicht ist ein **Prüfhinweis, keine Rechtsberatung**: eine konservative Heuristik (≥ 2 Muss-Kriterien oder hohes Restrisiko).

4.7 Mehrstufige Freigabe (Register „Freigabe")

Der Freigabe-Workflow richtet sich nach dem **Zielgruppenprofil** des Verfahrens:

Profil	Freigabe-Stufen
KMU	direkt der/die Datenschutzbeauftragte
Unternehmen	Fachbereich → Risikomanagement → Datenschutzbeauftragte(r)
Öffentliche Verwaltung	Fachbereich → Betriebsrat → Datenschutzbeauftragte(r)

Beim *Zur Prüfung geben* werden die Stufen erzeugt und der Reihe nach entschieden. **Jede Freigabe oder Ablehnung trägt eine elektronische Signatur** – einen manipulationssicheren Hash aus Entscheider, Zeitpunkt und Votum, der zusammen mit dem Vorgang im revisionssicheren Nachweis landet. Die **letzte** Stufe gibt das Verfahren frei; eine Ablehnung wirft es auf *Entwurf* zurück. Die Stufen bleiben als Nachweis stehen.

Nachweis – jede Änderung als Gegenüberstellung *alt* → *neu*, mit Benutzer und Zeitpunkt. Die Einträge sind unveränderbar und lassen sich nicht löschen; sie überleben sogar das Löschen des Verfahrens. Wer ein Verfahren entfernt, tilgt damit nicht den Nachweis, dass es existierte (Art. 5 Abs. 2).

4.8 Technisch-organisatorische Maßnahmen (Register „TOM")

Nach **Art. 30 Abs. 1 lit. g DSGVO** gehört eine allgemeine Beschreibung der technischen und organisatorischen Maßnahmen (TOM) in das Verzeichnis. Im Register **„TOM"** verknüpfen Sie die für das Verfahren getroffenen Maßnahmen aus einem **Katalog** (z. B. Zutritts-, Zugangs- und Zugriffskontrolle, Verschlüsselung, Pseudonymisierung, Verfügbarkeit, Auftrags- und Trennungskontrolle). Der Katalog wird mit Standard-TOMs ausgeliefert und lässt sich je Mandant erweitern. Die verknüpften Maßnahmen erscheinen im behördenkonformen Export (Abschnitt 7).

Lebenszyklus

Entwurf → *Zur Prüfung* → *Freigegeben*. Die Freigabe setzt zugleich die nächste Wiedervorlage. **Eine inhaltliche Änderung an einem freigegebenen Verfahren setzt die Freigabe zurück** – sonst zeigte das Verzeichnis einen freigegebenen Stand, den nie jemand freigegeben hat.

5 Kataloge (Betroffene und Datenkategorien)

Kategorien werden **zentral gepflegt** statt je Verfahren frei eingetippt – sonst stünden „Beschäftigte", „Mitarbeiter" und „Angestellte" nebeneinander, und der Export zeigte drei Gruppen, wo es eine gibt.

Ordivis Platform liefert den **gesetzlichen Grundkatalog** mit, weil er Gesetzestext und keine Konfiguration ist: die **besonderen Kategorien nach Art. 9 Abs. 1** vollständig (Gesundheit, Religion, Gewerkschaft,

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Biometrie, …), die **strafrechtlichen Daten nach Art. 10** getrennt davon, dazu die gängigen Betroffenengruppen. Diese Einträge sind änderbar, aber **nicht löschbar**; ebenso wenig lassen sich Kategorien löschen, die noch in einem Verfahren verwendet werden.

> **Art.-9-Daten** brauchen **zusätzlich** zur Art.-6-Grundlage einen Erlaubnistatbestand nach Art. 9 Abs. 2. Ordivis Platform trägt ihn bewusst **nicht** vor: Er hängt am Verfahren, nicht an der Kategorie – eine Vorbelegung wäre eine Rechtsbehauptung. Solange er fehlt, weist der Export ihn als Befund aus.

6 Meldewesen: Datenschutzverletzungen (Art. 33/34)

Datenschutz → *Datenpannen*. Die Liste zeigt als Erstes die **Restzeit** – denn Artikel 33 gibt Ihnen **72 Stunden**.

Die Frist läuft ab dem Bekanntwerden beim Verantwortlichen – nicht ab dem Vorfall und nicht ab der Erfassung im System. Ein Vorfall von vor zehn Tagen, der Ihnen vor einer Stunde bekannt wurde, hat noch 71 Stunden. Das Erfassungsformular führt diese drei Zeitpunkte deshalb getrennt.

Der Ablauf: **erfassen** (das darf jeder mit dem Melderecht – eine Meldepflicht, die am Recht scheitert, ist keine) → **bewerten** → **melden** → **abschließen**.

- **Bewertung**: kein Risiko / Risiko (Meldung an die Aufsichtsbehörde) / hohes Risiko (zusätzlich Benachrichtigung der Betroffenen nach Art. 34).
- **„Nicht meldepflichtig" verlangt eine Begründung.** Artikel 33 Abs. 5 verlangt die Dokumentation **aller** Verletzungen, auch der nicht gemeldeten – ohne Begründung ist die Entscheidung gegenüber der Behörde nicht haltbar.
- **Abschließen ist gesperrt**, solange eine Pflicht offen ist: meldepflichtig und nicht gemeldet, oder hohes Risiko ohne Benachrichtigung der Betroffenen. Eine offene Pflicht verschwindet nicht durch einen Klick.
- Die Überfällig-Markierung verschwindet **nicht durch Zeitablauf**, sondern erst durch die Meldung – eine verspätete Meldung ist weiterhin zu erstatten.

Jede Verletzung erhält ein fortlaufendes Aktenzeichen (`DSV-JJJJ-NNNN`), auf das die Aufsichtsbehörde sich beziehen kann.

7 Behördenkonformer Export (Art. 30 Abs. 4)

> 🌐 **Nur im Portal:** *Mein Arbeitsplatz* ▸ *Datenschutz-Zentrale* ▸ *Behördenkonformer Export* · 🔑 `datenschutz.vvt.approve` — Im Client führen Sie das Verzeichnis unter *Datenschutz* ▸ *Verarbeitungsverzeichnis*; der behördenkonforme Export liegt in der Zentrale des Portals.

1. Öffnen Sie *Mein Arbeitsplatz* ▸ *Datenschutz-Zentrale*.
2. Klicken Sie auf *Behördenkonformer Export*.
3. Wählen Sie das Format: **PDF** (Prüfunterlage), **CSV** (Weiterverarbeitung) oder **JSON** (maschinenlesbar).
4. Erzeugen Sie den Export und speichern Sie ihn ab.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

✅ **Ergebnis:** Sie erhalten das Verzeichnis mit Ihrem Mandanten als Verantwortlichem im Briefkopf – und mit den offenen Befunden ausgewiesen.

Der Export weist die Befunde mit aus. Das ist Absicht: Ein Export, der Lücken kaschiert, hilft in der Prüfung niemandem – die Behörde findet sie ohnehin, nur später und schlechter erklärt. Ausgewiesen werden unter anderem berechtigtes Interesse ohne Abwägung, Art.-9-Kategorien ohne Erlaubnistatbestand, Empfänger im Drittland ohne belegte Garantie sowie Verfahren ohne Rechtsgrundlage, ohne Empfängerkategorien, ohne Frist oder ohne Freigabe.

> **Was der Export abdeckt:** Artikel 30 Absatz 1 Buchstabe a bis f. Die allgemeine Beschreibung der technischen und organisatorischen Maßnahmen (Buchstabe g) ist noch nicht enthalten; das Gesetz formuliert sie mit „wenn möglich", die Vorlage scheitert daran nicht. Ergänzen Sie sie bis dahin im Anschreiben.

8 Fachbereiche melden selbst (Portal)

> 🌐 **Nur im Portal:** *Mein Arbeitsplatz ▸ Verarbeitung melden* · 🔑 `datenschutz.vvt.edit` — Der Client hat keinen Melde-Assistenten; dort führt der DSB das Verzeichnis unmittelbar.

Ein Datenschutzbeauftragter kann nicht alle Verarbeitungen eines Hauses allein kennen. Ein **Assistent** führt die Fachbereiche in fünf Schritten durch die Meldung – in Alltagssprache, ohne Datenschutz-Vorwissen:

🔒 **Pflichtreihenfolge** — der Assistent lässt Sie nicht weiter, solange eine Angabe fehlt, die er selbst abfragt. Das ist Absicht: Er soll keine halben Meldungen erzeugen.

1. **Worum geht es?** (Bezeichnung, Zweck in eigenen Worten)
2. **Wessen Daten verarbeiten Sie?**
3. **Welche Daten verarbeiten Sie?** – besonders schutzwürdige Daten sind gekennzeichnet
4. **Warum dürfen Sie das?** – bei „berechtigtem Interesse" wird direkt nach der Abwägung gefragt
5. **Wie lange behalten Sie die Daten?**

✅ **Ergebnis:** Die Meldung geht als **Entwurf zur Prüfung an den Datenschutzbeauftragten**; der Fachbereich gibt **nicht** selbst frei. Den eigenen Stand sehen die Meldenden danach unter *Mein Arbeitsplatz ▸ Meine Verfahren*.

> **Was der Assistent bewusst nicht fragt:** die **Empfänger** (Abschnitt 4.4). Ob eine Weitergabe an einen Auftragsverarbeiter, einen eigenverantwortlichen Dritten oder eine Behörde geht – und welche Garantie ein Drittland trägt – ist eine rechtliche Einordnung, keine Frage an den Fachbereich. Der DSB ergänzt sie bei der Prüfung; bis dahin trägt das Verfahren den Befund „keine Empfängerkategorien". Das ist gewollt: Der Entwurf sieht dann unfertig aus, weil er es ist.

Der Stand der eigenen Meldungen ist im Portal unter *Compliance & Sicherheit → Meine Verfahren* einsehbar: Der Fachbereich sieht seine gemeldeten Verarbeitungen mit Status (in Prüfung, freigegeben)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

und dem Hinweis „Angaben unvollständig", solange eine Pflichtangabe fehlt – aber **nur die eigenen**, nie fremde Verfahren, und ohne die Möglichkeit zur Freigabe (die bleibt der DSB-Zentrale vorbehalten).

9 Grenzen (Stand dieser Fassung)

Neu in dieser Fassung enthalten: die **Schwellenwertanalyse und Datenschutz-Folgenabschätzung** (Art. 35, Abschnitt 4.6), die **Verknüpfung mit CMDB-Systemen** samt Schutzbedarf (Abschnitt 4.5), der **mehrstufige, elektronisch signierte Freigabe-Workflow** (Abschnitt 4.7), der **Katalog technischer und organisatorischer Maßnahmen** (Art. 30 Abs. 1 lit. g, Abschnitt 4.8) sowie das **DSB-Lagebild** mit Kennzahl-Kacheln in der Zentrale.

Bewusst noch **nicht** vollständig im Produkt und für spätere Fassungen vorgesehen: Verträge zur Auftragsverarbeitung (Art. 28) sind auf API-Ebene angelegt, aber noch ohne eigene Oberfläche; die Verknüpfung der Verfahren mit **Geschäftsprozessen** fehlt noch.

> Das Modul ersetzt keine datenschutzrechtliche Beratung. Es führt das Verzeichnis, weist Lücken aus und belegt Änderungen – die fachliche Bewertung bleibt bei Ihrem Datenschutzbeauftragten.

Verwandte Themen

- **ID-28 – Modul Informationssicherheit (ISMS)** – technische und organisatorische Maßnahmen im ISMS
- **ID-21 – Modul CMDB** – die Systeme hinter einer Verarbeitungstätigkeit
- **ID-10 – Benutzerhandbuch – Self-Service-Portal** – Betroffenenrechte aus Sicht der Benutzer
- **ID-44 – Rollen- & Rechte-Matrix** – Rechte für Einsicht und Pflege

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · ORGANISATION

Modul Organisation (Ablauf, Aufbau, Geschäftsverteilungsplan)

Ablauf- und Aufbauorganisation, Produktkatalog und Geschäftsverteilungsplan

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-33 Organisation

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Organisation (Ablauf, Aufbau, Geschäftsverteilungsplan)

Der Geschäftsverteilungsplan ist in der öffentlichen Verwaltung das rechtlich bindende Dokument dafür, **wer für welche Aufgabe zuständig ist**. Er entsteht üblicherweise in einer Textverarbeitung – und veraltet ab dem Tag der Unterschrift: Die Amtsleitung wechselt, die Zuständigkeit im Dokument nicht.

Das Modul **Organisation** löst das an der Wurzel. Statt Zuständigkeiten je Aufgabe zu pflegen, wird **ein** Pflegepunkt gesetzt – die Zuordnung eines Produkts zu einer Organisationseinheit – und daraus die Federführung für alle Prozesse dieses Produkts abgeleitet. Bei einem Wechsel in der Amtsleitung wird das Organigramm angefasst, nicht 1.718 Prozesse.

> **Für wen gilt was?** *Ablauforganisation* und *Aufbauorganisation* stehen **allen** Mandanten zur Verfügung – Prozesslandkarte und Organigramm sind für Unternehmen genauso sinnvoll. *Produktkatalog* und *Geschäftsverteilungsplan* erscheinen nur, wenn die Organisationsart des Mandanten auf **Öffentliche Verwaltung** steht (wie die übrigen Verwaltungs-Zusatzfelder).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

1 Die Bausteine
2 Katalog ist nicht Bestand
2.1 Eigene Einträge und ein neuer Katalogstand
2.2 Quellenvermerk
2.3 Rechtsgrundlagen und das Bundesland
3 Ablauforganisation
4 Aufbauorganisation
4.1 Einheiten anlegen und ändern
4.2 Musterorganisation als Startpunkt
4.3 Personen eingeben
4.3.1 Die Vertretung gehört zur Stelle
4.4 Dieselbe Auskunft im Web-Portal
5 Geschäftsverteilungsplan
5.1 Zuständigkeiten zuordnen
5.2 Woher eine Zuständigkeit kommt
5.3 Zwei Sichten
5.4 Nachpflege
5.5 Prüfbericht
5.6 Fassungen und PDF
6 OZG-Leistungskatalog
6.1 Was diese Verwaltung überhaupt betrifft
6.2 Zuordnung zu den eigenen Prozessen
6.3 Umsetzungsstand bewerten
6.4 Digitalisierungsstand
6.5 Texte und Stammprozesse des Bundes
7 Rollen und Rechte
8 Reihenfolge in der Praxis
Verwandte Themen

1 Die Bausteine

Baustein	Was es ist
Produktkatalog	Der kommunale Produktplan der KGSt als Referenz: 6 Hauptbereiche, 16 Produktbereiche, 41 Produktgruppen, 148 Produkte
Prozesskatalog	5.019 Verwaltungsprozesse über 462 Fachaufgaben, je Prozess ein Produkt
Ablauforganisation	Die Sicht auf die Prozesslandschaft – als Landkarte und als Baum
Geschäftsverteilungsplan	Erzeugt aus Aufbauorganisation × Produkt × Prozess, nachpflegbar, als Fassung einfrierbar

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Baustein	Was es ist
OZG-Leistungskatalog	Der Umsetzungskatalog nach dem Onlinezugangsgesetz: 18 Themenfelder, 54 Lebens- und Geschäftslagen, 575 Leistungen – verbunden mit Produkt- und Prozesskatalog (Kapitel 6)

2 Katalog ist nicht Bestand

Der Produktplan und die 5.019 Prozesse liegen als **Referenz** in der Datenbank – nicht als Konfigurationselemente. Das ist eine bewusste Entscheidung: 5.019 Prozesse als CI wären 5.019 CI-Nummern, Audit-Zeilen und Suchtreffer für Material, das eine Kommune zu rund einem Drittel nie ausführt.

Erst die **Übernahme** erzeugt Geschäftsprozess-CIs. Sie erreichen sie unter *Organisation → Produktkatalog*:

1. Im Baum ein Produkt wählen oder in der Liste einen einzelnen Prozess.
2. **Vorschau** zeigt, was angelegt und was übersprungen würde – ohne etwas zu speichern.
3. **Übernehmen** legt die Prozesse an. Der Vorgang ist wiederholbar: Ein zweiter Lauf legt nichts erneut an.

Prozesse, die Ihre Verwaltung nicht wahrnimmt, kennzeichnen Sie mit **Nicht wahrnehmen**. Sie verschwinden aus Auswahl und Prüfbericht, ohne aus dem Katalog gelöscht zu werden.

> Für die Übernahme genügt das Recht zur Katalogpflege **nicht**. Weil dabei Bestand entsteht, ist zusätzlich `cmdb.ci.create` nötig – wer den Katalog pflegen darf, soll damit nicht ungewollt tausend Objekte in die CMDB schreiben.

2.1 Eigene Einträge und ein neuer Katalogstand

Der ausgelieferte Produktplan hat den Stand **Februar 2019**; die KGSt schreibt ihn fort. Der Standwechsel ist deshalb der Normalfall, nicht der Bruch:

- **Eigenen Eintrag ergänzen** legt ein Produkt oder eine Fachaufgabe unter dem gewählten Knoten an. Der Eintrag gehört Ihrem Mandanten und **überlebt den Standwechsel**.
- **Katalogstand** wechselt auf eine andere Ausgabe. Ihre Übernahmen und Ausschlüsse wandern über den Prozessnamen mit; Prozesse, die es im neuen Stand nicht mehr gibt, behalten ihr CI und werden gemeldet.

2.2 Quellenvermerk

Der Produktplan steht unter der Lizenz **CC BY 3.0 DE**. Der Vermerk (Quelle KGSt, Lizenzlink, Verweis auf den Datensatz, Angabe der Veränderungen) erscheint deshalb überall dort, wo Katalogdaten sichtbar werden – in der Katalogansicht, in der Ablauforganisation und auf dem GVP-PDF. Er ist keine Höflichkeit, sondern Bedingung der Nutzung, und er steht als **Datum am Katalogstand**, damit er beim nächsten Stand stimmt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2.3 Rechtsgrundlagen und das Bundesland

> **Bitte einmal lesen, bevor Sie eine Rechtsgrundlage aus dem Katalog verwenden.** Der ausgelieferte Prozesskatalog nennt bei **1.318 seiner 5.019 Prozesse hessisches Landesrecht** – HGO, HBO, HSchG, HSOG und weitere. Der KGSt-Produktplan selbst bleibt neutral und schreibt „GO der Länder"; die konkreten Fundstellen sind erst bei der Aufbereitung hinzugekommen. Für eine Kommune außerhalb Hessens ist „§ 66 HGO" damit nicht bloß unpassend, sondern **falsch**.

Die übrigen rund drei Viertel der Prozesse nennen **Bundesrecht** (SGB, WHG, BauGB, BImSchG, KrWG …). Das gilt unverändert in allen Ländern und ist unproblematisch.

So kommen Sie zur richtigen Fundstelle:

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Module*, Karte *Bundesland* · 🔑 `tenant.manage` — die Einstellung setzt der Plattform-Super-Admin.

🔒 **Pflichtreihenfolge** — ohne hinterlegtes Bundesland kennt Schritt 2 keine Landesfundstelle und Schritt 3 nennt statt der Übersicht den Hinweis, wo das Land einzutragen ist.

1. Hinterlegen Sie das **Bundesland des Mandanten** unter *Einstellungen* ▸ *Module*: Wählen Sie oben den Mandanten aus, dann erscheint die Karte **Bundesland**. Ordivis Platform leitet einen Vorschlag aus der Postleitzahl des Mandantenprofils ab, den Sie mit **„Vorschlag übernehmen"** oder durch eigene Auswahl **bestätigen** müssen. Die Einstellung setzt der Plattform-Super-Admin.
2. Danach zeigt die Detailkarte eines Prozesses neben der Fundstelle der Quelle auch die, die **in Ihrem Land** gilt – mit amtlicher Bezeichnung und Abkürzung.
3. Den Überblick über alle Rechtsgebiete gibt die Schaltfläche **„Landesrecht"** im Produktkatalog: Sie nennt jedes Rechtsgebiet mit der Fundstelle Ihres Landes, kennzeichnet Vorschläge als solche, verlinkt die Quelle – und sagt, wie viele Katalogprozesse überhaupt vom Landesrecht abhängen. Ohne hinterlegtes Bundesland steht dort genau das, samt Hinweis, wo Sie es eintragen.

Warum der Vorschlag bestätigt werden muss: Die Leitbereiche der Post folgen der Logistik, nicht den Landesgrenzen. Rund 200 Bereiche greifen über eine Grenze. Wo sich zwei Länder einen Bereich teilen, sagt Ordivis Platform ausdrücklich **„nicht festgelegt"** und schlägt gar nichts vor – ein plausibel aussehender Fehlvorschlag wäre hier schädlicher als keiner, weil er in eine Rechtsgrundlage mündet.

Solange kein Land bestätigt ist, wird **keine** landesrechtliche Fundstelle behauptet. Die Detailkarte sagt dann, dass der Text aus einer hessischen Aufbereitung stammt und was zu tun ist.

> **Geprüft und Vorschlag sind unterschieden.** Von den 31 Rechtsgebieten sind **16 gegen eine länderübergreifende Quelle abgeglichen** – sie decken alle 16 Länder ab und nennen die Quelle, an der das geschehen ist. Die übrigen sind begründete Vorschläge und **als solche gekennzeichnet** – bitte vor einer verbindlichen Verwendung am Landesrechtsportal Ihres Landes nachsehen. Eine ungeprüfte Angabe, die wie eine geprüfte aussieht, wäre der schlimmere Fehler.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Geprüft (je 16 Länder): Kommunalverfassungsrecht, Kreisrecht, Bauordnungsrecht, Schulrecht, Polizei- und Ordnungsrecht, Straßenrecht, Brand- und Katastrophenschutz, Landesbeamtenrecht, Archivrecht, Verwaltungsverfahrensrecht, Wasserrecht, Denkmalschutz, Waldrecht, Haushaltsrecht des Landes, Gaststättenrecht, Umweltinformationsrecht.

Vorschlag, noch nicht abgeglichen (ebenfalls je 16 Länder): Kommunalwahlrecht, Datenschutz- und Informationsfreiheitsrecht, öffentlicher Gesundheitsdienst, Rettungsdienst.

Nur die hessische Fundstelle hinterlegt: Vermessung und Geoinformation, Geodateninfrastruktur, Jagd, Landes-Jugendhilferecht, Wohnraumförderung, Agrarverwaltung, Psychisch-Kranken-Hilfe, Betreuung und Pflege, Hygiene, Baustatistik, Wohnungsbindung. Hier sagt die Detailkarte ausdrücklich, dass für Ihr Land noch keine Fundstelle hinterlegt ist – behauptet wird nichts.

Damit ist an **allen 1.318 betroffenen Prozessen** erkennbar, dass ihre Rechtsgrundlage landesabhängig ist. Das war zuvor nur bei 1.169 der Fall; die übrigen 149 zeigten hessisches Recht, ohne es kenntlich zu machen.

> **Für die drei Stadtstaaten** gilt eine Besonderheit: Berlin, Bremen und Hamburg haben **keine Gemeindeordnung**, weil Gemeinde- und Landesebene zusammenfallen. Dort steht ein entsprechender Hinweis samt der einschlägigen Landesverfassung statt einer erfundenen Fundstelle.

3 Ablauforganisation

Organisation → Ablauforganisation zeigt die Prozesslandschaft in zwei Formen.

Die **Landkarte** ordnet die Prozesse in drei Bändern an – Managementprozesse oben, Kernprozesse in der Mitte, Unterstützungsprozesse unten. Eine Kachel steht für einen Produktbereich (Kernband) beziehungsweise für ein Produkt (die beiden anderen Bänder) und nennt die Anzahl. Ein Klick auf eine Kachel schränkt die Liste rechts ein.

> **Warum die Kachelebene je Band verschieden ist:** Management- und Unterstützungsprozesse hängen im gelieferten Katalog vollständig unter einem einzigen Produktbereich (*11 Innere Verwaltung*). Ein Layout, das überall auf Bereichsebene kachelt, erzeugt dort zwei Bänder mit je genau einer Kachel – also eine leere Aussage.

Der **Baum** führt über Produktbereich, Produktgruppe und Produkt gezielt zu einem Ast. Jede Ebene grenzt ein: Ein Produkt zeigt seine eigenen Prozesse, ein Bereich oder eine Gruppe alle Prozesse darunter. Über **Auswahl aufheben** kommen Sie zur vollen Liste der gewählten Verwaltungsebene zurück.

Die **Verwaltungsebene** ist auf *Gemeinde/Stadt* vorbelegt (1.718 Prozesse). Ohne diesen Filter stünden alle 5.019 Prozesse in der Ansicht – das ist keine Landkarte, sondern eine Tapete. Mit **Nur übernommene** sehen Sie ausschließlich Ihren eigenen Bestand.

Als PNG exportieren legt die Landkarte als Bild ab, etwa für ein Handbuch oder einen Aushang.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4 Aufbauorganisation

Organisation → Aufbauorganisation führt Organisationseinheiten, Stellen und Besetzungen.

4.1 Einheiten anlegen und ändern

Links steht der Organisationsbaum, darunter **Neu**, **Ändern** und **Löschen**. Beim Anlegen erben Sie die markierte Einheit als übergeordnete – so entsteht die Gliederung in einem Zug.

Die **Einheiten** selbst sind Konfigurationselemente der Klasse *Abteilung / Organisationseinheit*; ihre Hierarchie ergibt sich aus den vorhandenen Enthaltenseins-Beziehungen – dieselbe Wahrheit wie der Standortbaum. Der **Einheitentyp** unterscheidet Dezernat, Amt, Fachbereich, Fachgebiet, Sachgebiet und Stabsstelle (bzw. Bereich, Abteilung, Team).

Umhängen heißt: im Änderungsformular die übergeordnete Einheit wechseln. Eine Einheit unter ihre eigene Untereinheit zu hängen wird abgelehnt – das ergäbe einen Ring ohne Wurzel.

Löschen geht nur, wenn nichts mehr daran hängt: keine Untereinheiten, keine Stellen, keine Produktzuordnungen. Andernfalls nennt die Meldung, was noch offen ist. Das ist Absicht – die Fremdschlüssel würden all das sonst kommentarlos mitlöschen und ganze Zweige des Geschäftsverteilungsplans verschwinden lassen.

4.2 Musterorganisation als Startpunkt

Eine leere Aufbauorganisation ist kein neutraler Anfang, sondern eine Sackgasse: keine Einheiten, also keine Stellen; keine Leitungsstelle, also keine Federführung. **Musterorganisation einspielen** (Registerkarte *Organigramm* oder der Hinweis oben) legt deshalb den Zuschnitt einer Kommune an: Verwaltungsleitung, Stabsstellen, fünf Fachbereiche mit ihren Fachgebieten, angegliederte Einrichtungen, je Einheit eine Leitungsstelle und die üblichen Beauftragtenfunktionen.

Alles davon ist **änderbar** – umbenennen, umhängen, löschen, ergänzen. Bei einem Mandanten, der schon Einheiten führt, fragt das Produkt nach und ergänzt dann nur, was fehlt; gleichnamige Einheiten und Stellen bleiben unverändert. Wer die Musterstruktur umbaut, bekommt sie beim nächsten Start **nicht** zurück.

> Die Stellen entstehen **unbesetzt**. Wer welche Stelle innehat, ist eine Angabe über einen konkreten Menschen und wird nicht mitgeliefert – das tragen Sie selbst ein.

Darunter liegt die **Stelle**. Sie ist die Ebene, die den Wechsel der Person überdauert – und genau darauf stützt sich der Geschäftsverteilungsplan. Eine Stelle trägt:

- ein **Stellenzeichen**, wie es im GVP erscheint (z. B. `32-1`),
- einen **Stellenumfang** in Vollzeitäquivalenten,
- ein **Leitungskennzeichen**: Genau eine Stelle je Einheit ist die Leitung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Warum die Leitung eine Stelle ist und kein Feld am Datensatz: Ein Personenverweis kann nicht abbilden, was in der Verwaltung Alltag ist – eine vakante, eine geteilte oder eine vertretene Leitung. Das frühere Feld *Leitung* an der Organisationseinheit entfällt deshalb; vorhandene Einträge werden beim ersten Start automatisch in eine Leitungsstelle mit Besetzung überführt.

Die **Besetzung** verbindet Stelle und Person, mit Anteil und Gültigkeit. Zwei Personen zu je 50 % sind zwei Besetzungen – die Summe darf 100 % nicht überschreiten. Eine **Vertretung** zählt nicht mit: Sie greift, wenn die Stelle nicht besetzt oder die Person abwesend ist.

Organisatorische Rollen sind Beauftragtenfunktionen (Informationssicherheits-, Datenschutz-, Brandschutzbeauftragte). Sie sind ausdrücklich **nicht** dasselbe wie eine Berechtigungsrolle (was jemand in der Software darf) oder eine Krisenstabsrolle im Notfallmanagement. Verweise zwischen diesen drei Dingen sind erlaubt, eine gemeinsame Ablage nicht – sonst entfernt der Entzug einer Softwareberechtigung jemanden aus dem Krisenstab.

Eine Rolle trägt **wer sie wahrnimmt** und **wer vertritt** – beides in einem Eintrag.

Warum die Vertretung hier ein Feld ist und bei der Stelle ein eigener Datensatz: Eine Stelle wird besetzt, mit Anteil, Zeitraum und Historie; die Vertretung ist dort eine zweite Besetzung. Eine Beauftragtenfunktion wird dagegen *bestellt*. „Informationssicherheitsbeauftragte: Meier, Vertretung: Schulz" ist eine einzige Aussage – auf zwei Zeilen verteilt entstünde nur die Frage, welche der beiden die Rolle ist, und der Geschäftsverteilungsplan zeigte eine Funktion zweimal.

Dieselbe Person kann sich nicht selbst vertreten; der Eintrag wird abgewiesen.

4.3 Personen eingeben

Jedes Personenfeld – Besetzung einer Stelle, Wahrnehmung und Vertretung einer Rolle – schlägt ab dem zweiten Buchstaben passende Namen vor. Vorgeschlagen wird aus drei Quellen, erkennbar am Zusatz hinter dem Namen:

Zusatz	Herkunft
Benutzer	ein Benutzerkonto dieses Mandanten
Person	ein Konfigurationselement der Klasse *Person (intern)*
Organisation	ein Name, der in dieser Aufbauorganisation bereits vorkommt

Die dritte Quelle trägt den Alltag: Wer seine Organisation gerade erst eingespielt oder importiert hat, führt seine Beschäftigten zunächst nur im Organigramm – Konten und Personen-CIs gibt es dafür meist nicht. Ein einmal eingetragener Name steht damit ab sofort überall zur Auswahl, und Schreibweisen bleiben einheitlich.

Freie Eingabe bleibt erlaubt. Wer tippt, statt zu wählen, bekommt denselben Eintrag wie zuvor. Eine neue Kollegin steht am ersten Tag im Organigramm und hat noch kein Konto – ein Auswahlzwang würde genau

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

dann blockieren. Wird ein Vorschlag *gewählt*, merkt sich der Eintrag zusätzlich die dahinterliegende Identität; weitergetippt gilt diese Zuordnung nicht mehr.

Nichts wird gelöscht. Eine aufgelöste Stelle und eine beendete Besetzung werden zum Stichtag beendet; mit **Historie anzeigen** sehen Sie sie weiterhin. Alte GVP-Fassungen bleiben dadurch lesbar.

4.3.1 Die Vertretung gehört zur Stelle

Im Dialog **Stelle anlegen** und **Stelle ändern** gibt es den Haken **Vertretung für diese Stelle**. Ist er gesetzt, erscheint darunter ein Suchfeld für die vertretende Person — dasselbe Personenfeld wie oben beschrieben.

> **Warum an dieser Stelle.** Die Vertretung ist im Datenmodell eine zweite Besetzung derselben Stelle. Erreichbar war sie nur über einen zweiten Schritt (*Stelle besetzen* mit gesetztem Haken „Vertretung"). Wer das nicht fand, legte stattdessen eine eigene Stelle „Stellv. …" an — mit dem Ergebnis, dass beide Stellen existierten, aber **nichts voneinander wussten** und die Spalte *Vertretung* leer blieb. Die Frage wird jetzt dort gestellt, wo die Stelle entsteht.

Beim **Wechsel** der Vertretung wird die bisherige zum heutigen Tag beendet und die neue begonnen. Der Name wird nicht überschrieben: Sonst hätte die neue Person rückwirkend vertreten, und die Historie wäre falsch. Wird der Haken entfernt, endet die Vertretung — sie bleibt in der Historie sichtbar.

4.4 Dieselbe Auskunft im Web-Portal

Unter *Organisation* zeigt das Web-Portal dieselben Einheiten und Stellen **lesend** – mit Inhaber und Vertretung, geteilte Besetzungen mit ihrem Anteil, und unbesetzte Stellen eigens gekennzeichnet. Genau das ist die Frage, die von unterwegs gestellt wird: *Wer macht das gerade, und wer vertritt?* Gepflegt wird die Aufbauorganisation weiterhin nur im Arbeitsplatz-Client.

> **Hinweis:** Bis `2026.08.08.1302` blieb die Stellenspalte im Portal leer und jede Stelle war als *unbesetzt* ausgewiesen – auch die besetzten. Sehen Sie dieses Bild noch, läuft das Portal auf einem älteren Stand.

5 Geschäftsverteilungsplan

5.1 Zuständigkeiten zuordnen

Unter *Organisation* → *Geschäftsverteilungsplan* → *Zuständigkeiten* verbinden Sie Produkte mit Organisationseinheiten. Aus dieser einen Zuordnung folgt die Federführung für **alle** Prozesse des Produkts.

Startbelegung übernehmen wertet den Ämtervorschlag des Prozesskatalogs aus und legt Zuordnungen an, wo es eine gleichnamige Organisationseinheit gibt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Der Vorschlag ist ein Vorschlag. Die Spalte des Katalogs ist je Produktbereich konstant und beschreibt einen üblichen Ämterschnitt – jede Kommune schneidet ihre Ämter anders. Solche Zuordnungen sind als *Vorschlag* gekennzeichnet, bis jemand sie mit **Vorschlag bestätigen** übernimmt. Wer sie für die reale Aufbauorganisation hält, baut den Plan auf Sand.

5.2 Woher eine Zuständigkeit kommt

Die Federführung („A" der RACI-Matrix) wird nach dieser Rangfolge bestimmt:

Rang	Regel	Anzeige
1	ausdrücklicher RACI-Eintrag am Prozess	**gesetzt**
2	Produkt des Prozesses → Zuordnung → Leitung der Einheit	**geerbt von ⟨Produkt⟩**
3	nächsthöherer Katalogknoten mit Zuordnung, sonst nächsthöhere Einheit mit Leitung	**geerbt von ⟨…⟩**
4	keine Regel greift	**Fehlstelle** – erscheint im Prüfbericht

Die **Herkunft steht überall dabei** – in der Liste, im Detail und im PDF. Ein geerbtes A, das wie ein gepflegtes aussieht, führt zu falschen Zuständigkeitsauskünften, und genau dafür ist ein GVP da.

Mitwirkung (R), Beratung (C) und Information (I) erben umgekehrt **nach unten**: Sie lassen sich auf Produkt-, Fachaufgaben- oder Prozessebene zuweisen, und die speziellere Zuweisung sticht. Ohne das wäre die Matrix bei 1.718 Prozessen nicht pflegbar.

5.3 Zwei Sichten

Der Schalter oben rechts wechselt zwischen **Leitungssicht** (alles, wofür eine Einheit die Federführung hält) und **Sachbearbeitungssicht** (nur das, woran die eigene Stelle als R oder C beteiligt ist).

5.4 Nachpflege

Nicht jede Zuständigkeit folgt aus Produkt und Prozess. **Sonderregelung ergänzen** trägt einen freien Eintrag unter einer Einheit nach – mit **Begründung**. Jeder solche Eintrag ist im Plan und im PDF als **Abweichung** markiert: Sonst ist nach zwei Jahren nicht mehr erkennbar, was aus den Daten folgt und was jemand von Hand hineingeschrieben hat.

5.5 Prüfbericht

Der Reiter **Fehlstellen** nennt genau die fünf Dinge, die einen Geschäftsverteilungsplan unbrauchbar machen:

1. Prozesse ohne Federführung,
2. doppelte Federführung an einem Prozess,
3. Produkte mit übernommenen Prozessen, aber ohne Zuordnung,
4. Einheiten ohne Leitung (eine geerbte Federführung endet dort),

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5. verwaiste Zuordnungen nach einer Umorganisation.

5.6 Fassungen und PDF

Entwurf als PDF erzeugt den Plan zum Gegenlesen. **Fassung einfrieren** erzeugt die verbindliche Ausgabe: vollständiger Datenstand, PDF und dessen **SHA-256-Prüfsumme**, mit Ausgabedatum und Ersteller.

> Erst die Fassung beweist etwas. Was ein GVP rechtlich leisten muss, ist der Nachweis *„diese Fassung galt am 15.03.2027"* – deshalb liest die Stichtagsabfrage die Fassung, statt den Stand zu rekonstruieren. Eine eingefrorene Fassung wird nicht mehr verändert; Änderungen ergeben eine neue Fassung.

6 OZG-Leistungskatalog

Das Onlinezugangsgesetz verpflichtet Bund, Länder und Kommunen, ihre Verwaltungsleistungen auch digital anzubieten. Der **Umsetzungskatalog** ordnet diese Leistungen nicht nach Behörden, sondern danach, in welcher Situation Menschen und Unternehmen sie brauchen: 18 Themenfelder, 54 Lebens- und Geschäftslagen, 575 Leistungen.

Er ist damit die **Außensicht** auf dieselbe Verwaltung, die Produktplan und Prozesskatalog von innen beschreiben. Beide Seiten sind im Modul verbunden – der Katalog erscheint unter *Organisation → OZG-Leistungen* und nur bei Mandanten der Organisationsart **Öffentliche Verwaltung**.

6.1 Was diese Verwaltung überhaupt betrifft

Nicht jede der 575 Leistungen wird von jeder Behörde vollzogen. Die Vollzugsebene steht in der LeiKa-Typisierung der jeweiligen Leistungskennung und wird als **Vollzugsebene** angezeigt:

Anzeige	Bedeutung
Bund	Die Leistung vollzieht eine Bundesbehörde
Land oder Kommune	Der Vollzug liegt beim Land *oder* bei der Kommune – welches von beidem, hängt vom jeweiligen Landesrecht ab
Kommune	Kommunaler Vollzug
unbestimmt	Die Quelle trifft keine Aussage

> **„Land oder Kommune" ist kein Anzeigefehler.** Die Typisierung des Bundes unterscheidet an dieser Stelle nicht weiter – welche Ebene zuständig ist, entscheidet das Landesrecht je Bundesland. Wer daraus eine kommunale Zuständigkeit ableitet, behauptet mehr als die Quelle hergibt. Der Filter **Nur was uns betrifft** blendet deshalb nur aus, was nachweislich eine andere Ebene betrifft; „Land oder Kommune" und „unbestimmt" bleiben immer sichtbar.

Die Verwaltungsebene Ihres Mandanten (Bund, Land, Kreis, Kommune) setzt die SuperAdministration unter *Plattform → Mandanten*. Ist sie nicht gesetzt, filtert der Katalog nichts weg – eine Vorauswahl aufgrund

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

einer nicht getroffenen Angabe sähe aus wie eine Entscheidung.

6.2 Zuordnung zu den eigenen Prozessen

Die Brücke zwischen OZG-Leistung und eigener Verwaltung ist der **LeiKa-Schlüssel**, den auch der Prozesskatalog führt. Er zeigt auf **ein** Produkt, aber auf bis zu 722 einzelne Prozesse – deshalb wird auf **Produktebene** verbunden. Das ergibt 235 belastbare Paare statt 18.784 kaum prüfbarer Einzelbezüge.

- **Abgeleitet** – aus dem Schlüsselvergleich entstanden, gilt für alle Mandanten, wird bei jedem Katalogstand neu gebildet.
- **Ergänzt** – von Hand über **Zuordnung ergänzen**, wenn die Ableitung eine Verbindung nicht findet.
- **Aufgehoben** – eine abgeleitete Zuordnung, die für diese Verwaltung nicht stimmt. Sie wird nicht gelöscht, sondern mit Begründung abgelehnt und ist im Prüfbericht nachvollziehbar.

Aus der Produktzuordnung folgt die Federführung: Wer im Geschäftsverteilungsplan für das Produkt zuständig ist, ist es auch für die OZG-Leistung. Ein zweiter Pflegepunkt entsteht nicht.

6.3 Umsetzungsstand bewerten

Bewerten hält je Leistung fest, wie weit die Digitalisierung ist:

Feld	Inhalt
OZG-Reifegrad	0 nur Information · 1 Formular zum Herunterladen · 2 Antrag online, Bearbeitung im Haus · 3 durchgängig online inkl. Bescheid · 4 Once-Only aus Registern
Umsetzungsart	Eigenentwicklung · EfA-Nachnutzung · EfA-Bereitstellung · über Portalverbund/Dienstleister · nicht zuständig
Fachverfahren	Das CI der Anwendung, die den Antrag verarbeitet
Online-Dienst	Die URL, unter der Bürger oder Unternehmen die Leistung erreichen

> **„Noch nicht bewertet" ist ein eigener Zustand, nicht Reifegrad 0.** Eine nicht getroffene Aussage als „nur Information" zu zählen, würde den Digitalisierungsstand systematisch zu gut aussehen lassen.

Ab Reifegrad 2 läuft der Antrag durch ein IT-System. Fehlt dann das Fachverfahren, meldet der **Prüfbericht** den Schnittstellenbruch – ebenso eine EfA-Nachnutzung ohne Angabe, wer den Dienst bereitstellt.

6.4 Digitalisierungsstand

Organisation → OZG-Digitalisierungsstand fasst zusammen: Reifegrade, Umsetzungsart, Vollzugsebenen, Zuordnungsgrad je Themenfeld, bewertete Leistungen je Organisationseinheit und die offenen Fehlstellen. Gezählt werden nur Leistungen, die diese Verwaltung vollzieht.

6.5 Texte und Stammprozesse des Bundes

Der Umsetzungskatalog beschreibt Leistungen knapp. Ausführliche, bundeseinheitliche Texte liefert das **FIM-Portal** – sie erscheinen im Detailbereich als *Bundeseinheitliche Beschreibung (LeiKa-plus)*: Volltext,

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Erforderliche Unterlagen, Fristen, Kosten, Rechtsgrundlagen. Sie sind mitgeliefert und brauchen keinen Internetzugang.

> Diese Texte hängen fachlich an der **Verrichtung** (z. B. „Erteilung"), nicht am Leistungsobjekt. Aufgelöst wird deshalb über die ersten acht Stellen der Kennung – sonst bliebe der Text bei fast jeder Leistung leer.

Stammprozesse holen ruft zusätzlich den vom Bund modellierten Referenzablauf aus der FIM-Prozessbibliothek ab (Internetzugang nötig, abschaltbar). Er zeigt Handlungsgrundlagen, Beteiligte, Auslöser und den Ablauf als BPMN-Modell und wird gegen Ihre freigegebene Prozessdefinition verglichen:

- **Ein fehlender Schritt ist ein Hinweis** – verglichen wird über die Benennung, der Schritt kann bei Ihnen schlicht anders heißen.
- **Eine fehlende Handlungsgrundlage ist eine Lücke** – das ist keine Frage der Benennung.
- **Eigene Zusatzschritte sind kein Befund.** Der Stammprozess zeigt die Rechtslage, nicht Ihre Hausorganisation; er ist eine Untergrenze, keine Schablone.

7 Rollen und Rechte

Recht	Erlaubt
`gvp.read`	Produktkatalog, Ablauf-/Aufbauorganisation, Geschäftsverteilungsplan und OZG-Leistungskatalog lesen
`gvp.manage`	Katalogpflege, Zuständigkeiten, Stellen und Besetzungen, Nachpflege, Fassungen einfrieren, OZG-Zuordnung und -Bewertung
zusätzlich `cmdb.ci.create`	Übernahme aus dem Katalog in den Bestand (dabei entstehen CIs)

Die Rolle **CMDB-Administration** bringt beide GVP-Rechte mit; die Rolle **CIO** erhält `gvp.read` für die Leitungssicht.

8 Reihenfolge in der Praxis

> 💻 **Client:** *Organisation* ▸ *Aufbauorganisation*, *Produktkatalog*, *Geschäftsverteilungsplan* · 🌐 **Portal:** *Organisation* ▸ *Aufbauorganisation* (nur lesend) · 🔑 `cmdb.ci.read`, zum Pflegen `cmdb.ci.update`

🔒 **Pflichtreihenfolge** — jeder Schritt setzt das Ergebnis des vorigen voraus. Ohne besetzte Leitungsstellen bleibt die Startbelegung in Schritt 4 leer, und ohne bestätigte Zuständigkeiten hat der Prüfbericht in Schritt 5 nichts zu prüfen. Diese Folge ist der eigentliche Inhalt des Kapitels.

1. Organisationseinheiten in der CMDB anlegen und über *übergeordnet setzen* hierarchisch verbinden.
2. Je Einheit eine **Leitungsstelle** anlegen und besetzen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3. Im **Produktkatalog** die Prozesse der eigenen Verwaltungsebene übernehmen; nicht Wahrgenommenes ausschließen.
4. Unter **Zuständigkeiten** die Startbelegung übernehmen und Produkt für Produkt bestätigen.
5. Den **Prüfbericht** abarbeiten, bis keine Fehlstelle mehr offen ist.
6. **Fassung einfrieren** und das PDF zur Unterschrift geben.
7. Im **OZG-Leistungskatalog** die Zuordnungen prüfen und den Umsetzungsstand bewerten – der Digitalisierungsstand ist erst aussagekräftig, wenn die Federführung steht.

✅ **Ergebnis:** Es liegt ein unterschriftsreifer Geschäftsverteilungsplan vor, dessen Fassung eingefroren und damit nachweisbar ist.

Verwandte Themen

- **ID-21a – Datenimport – Geschäftsprozesse** – Prozesse per Datei übernehmen statt einzeln anlegen
- **ID-29 – Modul Business Continuity Management** – Business-Impact-Analyse auf denselben Prozessen
- **ID-41 – Modul Identität, Benutzer & Rollen** – Personen hinter Federführung und Vertretung
- **ID-26 – Modul Workflows & Aufgaben** – Abläufe automatisieren

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · SELBSTORGANISATION

Modul Selbstorganisation (Aufgaben, Board, Kalender)

Persönliches Aufgabenbrett, Aufgabenliste und Kalender – mit Bezug zu Tickets, CIs und Änderungen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-34 Selbstorganisation

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Selbstorganisation (Aufgaben, Board, Kalender)

Zwischen den Vorgängen, die ein Ticketsystem kennt, und dem was tatsächlich auf dem Schreibtisch liegt, klafft in der Praxis eine Lücke: der Anruf, aus dem noch kein Ticket geworden ist; der Punkt aus einer Besprechung; die Erinnerung „nächste Woche nochmal nachfassen". Diese Dinge landen sonst auf Haftnotizen, in einer privaten Notiz-App oder im eigenen Postfach – also überall dort, wo sie beim Urlaub oder Ausscheiden verloren gehen.

Das Modul **Selbstorganisation** gibt dafür einen Platz *innerhalb* der Plattform. Es ist ausdrücklich **kein zweites Ticketsystem**: Aufgaben haben keine SLA, keinen Melder und keine Eskalation. Sie sind persönliche Merkposten – die sich aber mit einem Ticket, einem CI oder einer Änderung **verknüpfen** lassen und damit im Zusammenhang bleiben.

> **Sichtbarkeit.** Eigene Aufgaben sieht **nur die Person selbst**. Auch Administratoren sehen sie nicht. Wer die Arbeit eines Teams sehen will, benutzt das **Team-Board** (Abschnitt 6) – das zeigt bewusst weniger und beruht auf einer ausdrücklichen Berechtigung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

1 Die drei Sichten
2 Aufgabenbrett
Semantischer Zustand statt Spaltenname
WIP-Grenze
Spalten anpassen
3 Meine Aufgaben (Liste)
Anlegen, ändern, löschen
Verknüpfungen
4 Aufgaben aus dem Zusammenhang heraus anlegen
5 Mein Tag
6 Übergabe und Team-Board
Aufgaben übergeben
Team-Board
7 Mein Kalender
7.1 Wochenbeginn, Wochentage und Kalenderwochen
7.2 Wohin ein Klick führt
8 Erinnerungen und Kalender-Abonnement
Erinnerungen
ICS-Feed abonnieren
9 Exchange-Anbindung (optional)
Einrichtung (für Administratoren)
10 Unterschiede zwischen Client und Portal
11 Rollen und Rechte
12 Was dieses Modul ausdrücklich nicht tut
Verwandte Themen

1 Die drei Sichten

Das Modul erscheint in der Navigation als drei Einträge auf dieselbe Datenbasis:

Sicht	Wofür	Kapitelabschnitt
Aufgabenbrett	Kanban-Ansicht mit Spalten – der Blick auf den Fluss	2
Meine Aufgaben	Listenansicht mit Filter, Sortierung und Mehrfachauswahl – der Blick aufs Detail	3
Mein Kalender	Alles mit Datum an einer Stelle, modulübergreifend	7

Alle drei stehen sowohl im **Client** als auch im **Self-Service-Portal** zur Verfügung (Unterschiede: Abschnitt 10).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2 Aufgabenbrett

Das Brett wird beim ersten Aufruf automatisch mit fünf Spalten angelegt: *Backlog*, *Zu tun*, *In Arbeit*, *Wartet*, *Erledigt*. Karten werden im Client per **Ziehen und Ablegen** verschoben, in eine andere Spalte oder innerhalb einer Spalte an eine andere Position.

Semantischer Zustand statt Spaltenname

Jede Spalte trägt neben ihrem Namen einen **semantischen Zustand** (`Backlog`, `Todo`, `Doing`, `Waiting`, `Done`). Das ist der eigentliche Grund, warum Spalten frei benannt werden dürfen: Der Rest der Plattform – Erinnerungen, Kalender, Auswertungen – fragt nie nach dem Spaltennamen, sondern nach dem Zustand. Wer seine Spalte „Heute" statt „Zu tun" nennt, ändert damit die Beschriftung, nicht das Verhalten.

- Zu **jedem** Zustand muss mindestens eine Spalte existieren; genau eine davon ist die **Standardspalte** für diesen Zustand (dort landen Aufgaben, die ohne Spaltenangabe in diesen Zustand wechseln).
- Wird eine Spalte gelöscht, deren Zustand sonst keine Spalte mehr hätte, weist die Plattform das ab. Sonst gäbe es Aufgaben, die nirgendwo hinkönnen.
- Das Verschieben nach *Erledigt* setzt den Erledigungszeitpunkt; das Zurückholen löscht ihn wieder.

WIP-Grenze

Je Spalte lässt sich eine **WIP-Grenze** (Work in Progress) setzen. Wird sie überschritten, färbt sich die Zählung der Spalte – ⚠️ **sie blockiert das Ablegen aber nicht**. Das ist eine bewusste Entscheidung: Ein Brett, das die Annahme einer Karte verweigert, wird nicht eingehalten, sondern umgangen. Die Grenze soll ein Gespräch auslösen, keine Sperre sein.

Spalten anpassen

Über *Spalten bearbeiten* lassen sich Spalten anlegen, umbenennen, umsortieren, mit einer WIP-Grenze versehen und wieder entfernen. Die Anordnung wird als **ein** Vorgang gespeichert – nicht Spalte für Spalte –, damit ein halb gespeicherter Zwischenstand nicht kurzzeitig gegen die Regeln oben verstößt.

3 Meine Aufgaben (Liste)

Dieselben Aufgaben als vollwertige Tabelle: Filter, Sortierung, Spaltenwahl und Export wie in allen Tabellen der Plattform (siehe Kapitel *Bedienung*). Je Aufgabe lassen sich pflegen:

Feld	Bedeutung
Titel	Pflichtfeld – alles andere ist optional
Notizen	Freitext
Priorität	*Niedrig*, *Normal*, *Hoch*, *Dringend* – steuert nur den Farbstreifen der Karte
Fällig am	Zeitpunkt; überfällige Aufgaben erscheinen rot
Geplant für	Ein **Tag** ohne Uhrzeit – die Grundlage für *Mein Tag* (Abschnitt 5)
Aufwand	Geschätzte Minuten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Bedeutung
Verknüpfungen	Beliebig viele Bezüge zu Tickets, CIs, Änderungen, Problemen

⚠️ **Fällig** und **Geplant für** sind zwei verschiedene Dinge und absichtlich getrennt: *Fällig* ist eine Zusage nach außen, *Geplant für* eine Absicht nach innen. Wer beides in ein Feld presst, verliert die Möglichkeit, eine Aufgabe vorzuziehen, ohne ihre Frist zu verschieben.

Anlegen, ändern, löschen

Über der Tabelle stehen drei Schaltflächen: **Neu …** öffnet das leere Formular mit allen Feldern oben, **Bearbeiten …** öffnet die markierte Aufgabe – ein **Doppelklick** auf die Zeile tut dasselbe –, und **Löschen …** entfernt die markierten Aufgaben nach einer Rückfrage. Mehrere Zeilen lassen sich dafür zugleich markieren; die Rückfrage nennt dann ihre Anzahl.

Das Eingabefeld links daneben bleibt die **Schnellerfassung**: Titel eintippen, Eingabetaste, fertig – das Übrige wird später ergänzt. Beide Wege legen dieselbe Aufgabe an; die Schnellerfassung setzt lediglich nichts außer dem Titel.

Ein Wechsel der **Spalte** im Formular verschiebt die Aufgabe ans Ende der Zielspalte. Die Reihenfolge *innerhalb* einer Spalte bleibt Sache des Bretts – sie ist eine Aussage darüber, was zuerst drankommt, und die trifft man beim Ziehen, nicht in einem Formular.

Verknüpfungen

Eine Verknüpfung zeigt den **Namen** des verknüpften Objekts, nicht seine Id – und bei Tickets zusätzlich die Vorgangsnummer. Fehlt der Name, weil das Zielobjekt gelöscht wurde oder die Berechtigung fehlt, steht dort ein **Hinweis** statt eines leeren Feldes: Ein leeres Namensfeld sähe aus wie „nicht verknüpft" und wäre damit eine falsche Aussage.

Im Bearbeitungsformular steht hinter jeder Verknüpfung ein Kreuz, das sie **löst**. Wirksam wird das erst mit *Speichern* – wer abbricht, hat nichts gelöst. **Angelegt** werden Verknüpfungen nicht hier, sondern am Objekt selbst (Abschnitt 4): Dort steht der Bezug, hier stünde nur eine Liste von Kennungen.

4 Aufgaben aus dem Zusammenhang heraus anlegen

Der übliche Weg ist nicht das leere Formular, sondern das Kontextmenü an anderer Stelle. In Ticket-, CI- und Änderungslisten steht **Als eigene Aufgabe merken**: Titel und Verknüpfung werden übernommen, die Aufgabe landet in der Standardspalte für *Zu tun*.

Das ist der eigentliche Sinn des Moduls – ein Merkposten entsteht dort, wo der Gedanke entsteht, ohne Wechsel der Ansicht.

5 Mein Tag

Mein Tag ist kein eigener Datenbestand, sondern eine **Auswahl**: alles, was für heute geplant ist (*Geplant für* = heute), plus alles, was heute oder früher fällig ist und noch nicht erledigt wurde. Aufgaben lassen sich

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

mit einem Klick **auf heute ziehen** oder **auf morgen schieben**.

Der Sinn: Die Frage „was mache ich als Nächstes?" wird einmal am Tag beantwortet, nicht bei jedem Blick auf das Brett neu.

6 Übergabe und Team-Board

Aufgaben übergeben

Eine Aufgabe lässt sich einer anderen Person **anbieten**. Sie wechselt dabei nicht sofort den Besitzer, sondern durchläuft drei Zustände: *angeboten* → *angenommen* oder *abgelehnt*.

- Erst mit dem **Annehmen** wandert die Aufgabe auf das Brett der anderen Person – in deren Standardspalte, nicht in eine Spalte, die es dort vielleicht gar nicht gibt.
- Beim **Ablehnen** bleibt sie, wo sie war; die Ablehnung wird mit Begründung protokolliert.
- Eine bereits entschiedene Übergabe lässt sich nicht ein zweites Mal entscheiden.
- Beide Seiten werden benachrichtigt (`selforg.handover.offered`, `selforg.handover.decided`).

⚠️ **Ohne Annahme keine Übergabe.** Eine Aufgabe einfach auf ein fremdes Brett zu legen, wäre eine Zuweisung von Arbeit ohne Zustimmung – dafür ist das Ticketsystem zuständig, nicht das persönliche Brett.

Team-Board

Wer `selforg.board.team.read` besitzt, sieht ein **Team-Board**: die Aufgaben mehrerer Personen nebeneinander, in denselben Spalten. Es zeigt **Titel, Spalte und Fälligkeit** – keine Notizen.

Die Mitglieder eines Team-Boards kommen aus einer von zwei Quellen:

1. einer **Supportgruppe** des Service Desk, oder
2. einer **Organisationseinheit** aus dem Organisationsmodul (Kapitel *Organisation*).

⚠️ Bei der Organisationseinheit gelten drei Regeln, die häufig überraschen:

- **Kein Aufstieg im Baum.** Eine übergeordnete Einheit sieht nicht automatisch die Aufgaben der untergeordneten. Wer das will, wählt die untergeordnete Einheit ausdrücklich hinzu.
- **Vertretung nur bei unbesetzter Stelle.** Eine hinterlegte Vertretung wird genau dann Mitglied, wenn die Stelle selbst **nicht** besetzt ist. Sonst wären während einer Vertretung beide sichtbar.
- **Zeitliche Gültigkeit.** Zuordnungen mit Beginn- und Enddatum zählen nur innerhalb ihres Zeitraums. Eine im Vorgriff erfasste künftige Besetzung wirkt erst ab ihrem Beginn.

7 Mein Kalender

Der Kalender ist eine **Sicht**, kein Speicher. Er sammelt aus den Modulen ein, was ein Datum hat:

Eintragsart	Herkunft
Fälligkeit, Wiedervorlage, Zurückstellung, Termin	Service Desk (Tickets)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Eintragsart	Herkunft
Eigene Aufgabe (fällig)	dieses Modul
Wartungsfenster, Änderungsfenster	CMDB / Änderungsmanagement
Workflow-Frist	Workflow
Termin bzw. *Belegt*	Exchange (Abschnitt 9)

Vier Ansichten im Client (Monat, Woche, Tag, Agenda), drei im Portal. Ein Klick auf einen Eintrag springt **in das Quellobjekt** – geändert wird ein Datum immer dort, wo es hingehört, nie im Kalender.

> ⚠️ **Wenn ein Modul nicht antwortet**, steht das als Warnung über dem Kalender. Diese Meldung ist wichtig: Ein leerer Kalender behauptet „nichts zu tun", und genau das wäre dann falsch.

7.1 Wochenbeginn, Wochentage und Kalenderwochen

Über den Spalten stehen die **Wochentage**, links neben jeder Zeile die **Kalenderwoche**. Absprachen laufen über KW-Nummern („liefern wir in KW 34"); ohne sie lässt sich der Kalender dagegen nicht halten.

So stellen Sie den Wochenbeginn um:

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Erscheinungsbild*, Karte *Kalender* · 🔑 kein Recht nötig – die Einstellung gilt nur für Ihr Konto — Das Portal übernimmt die Einstellung, bietet sie aber nicht zum Ändern an.

1. Öffnen Sie *Einstellungen* ▸ *Erscheinungsbild* und suchen Sie die Karte *Kalender*.
2. Wählen Sie unter *Wochenbeginn* zwischen *Montag* (Vorgabe, ISO 8601) und *Sonntag*.

✅ **Ergebnis:** Der Hinweis *Wirkt sofort im Kalender.* trifft zu – Spalten und Kalenderwochen-Nummern ordnen sich neu. Die Einstellung gilt je Benutzer und wird serverseitig mitgeführt, gilt also auch an einem anderen Rechner.

> **Die Zählweise der Kalenderwochen folgt dem Wochenbeginn.** Bei Montag gilt ISO 8601: Die erste Woche mit vier Tagen im neuen Jahr ist KW 1. Bei Sonntag gilt die im angelsächsischen Raum übliche Regel: Die Woche, in die der 1. Januar fällt, ist KW 1. Beides zu mischen wäre falsch — neben einer Zeile, die sonntags beginnt, stünde sonst eine Nummer, die eine montags beginnende Woche zählt.
>
> Ein Nebeneffekt der Sonntagsregel: Die Woche über den Jahreswechsel trägt zwei Nummern (KW 53 des alten und KW 1 des neuen Jahres). Das Raster zeigt die Nummer ihres ersten Tages. Nach ISO 8601 tritt das nicht auf, weil dort jede Woche ganz zu einem Jahr gehört.

Die Wochentagsnamen kommen aus der eingestellten Sprache des Clients; sie sind nicht gesondert zu pflegen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

7.2 Wohin ein Klick führt

Eintragsart	Sprungziel
Ticket-Fälligkeit, Wiedervorlage, Zurückstellung, Termin	Der Vorgang
Eigene Aufgabe	Das Aufgabenbrett
Wartungsfenster	Die Liste der Wartungsfenster, der Eintrag ist dort hervorgehoben
Änderungsantrag	Der Änderungsantrag
Termin aus Exchange	*Kein Sprungziel* — ein Hinweis nennt Art und Uhrzeit

Ein Klick, der nichts tut, sieht aus wie ein Fehler. Wartungsfenster und Änderungsanträge führten früher ins Leere: Sie tragen keine Ticketnummer, und die Zeile blieb stumm. Wer den Eintrag im Kalender sah, konnte ihn nirgends wiederfinden. Beide springen jetzt auf ihre Seite; was kein Ziel im Client hat — Termine aus Exchange liegen außerhalb —, sagt das ausdrücklich.

8 Erinnerungen und Kalender-Abonnement

Erinnerungen

Je Benutzer lässt sich einstellen, **wie lange vorher** an fällige Aufgaben erinnert wird (Vorgabe: 60 Minuten) und über welchen Kanal. Ein Hintergrundjob prüft das regelmäßig und meldet einmalig je Aufgabe (`selforg.task.due-soon`, `selforg.task.due`) – eine bereits gemeldete Aufgabe wird nicht erneut gemeldet, auch nicht nach einem Neustart.

ICS-Feed abonnieren

Unter *Mein Kalender → Abonnieren* entsteht eine **persönliche Feed-Adresse** im iCalendar-Format (RFC 5545). Sie lässt sich in Outlook, Thunderbird, Apple Kalender oder Android als Internetkalender eintragen.

- Die Adresse enthält ein **Token** – sie ist damit ein Geheimnis und ersetzt die Anmeldung. Wer sie weitergibt, gibt seinen Kalender weiter.
- Das Token wird **einmal** im Klartext angezeigt; danach ist es nur noch **widerrufbar**, nicht wiederherstellbar. Ein neues Token macht die alte Adresse sofort ungültig.
- Der Feed ist **ausschließlich lesend**. Ein Kalenderprogramm kann darüber nichts anlegen oder ändern.
- Wie oft aktualisiert wird, entscheidet das Kalenderprogramm – bei Outlook typischerweise stündlich. Das ist eine Eigenschaft des Programms, nicht der Plattform.

9 Exchange-Anbindung (optional)

Ist ein On-Premises-Exchange angebunden, erscheinen im Kalender zusätzlich die eigenen Exchange-Termine, und – wenn eingeschaltet – werden Fälligkeiten als Termine **nach** Exchange geschrieben.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Zwei Punkte, die für die Einführung entscheidend sind:

- **Betriebsart *Nur Frei/Gebucht*** (Vorgabe) zeigt belegte Zeiträume **ohne** Betreff, Ort und Teilnehmer. Für viele Personalvertretungen ist das der Unterschied zwischen zustimmungsfähig und nicht.
- **Jeder Benutzer gibt sein Postfach selbst frei** (*Mein Konto → Postfächer*). Ohne diese Freigabe greift die Plattform auf kein Postfach zu – auch nicht auf das eines Administrators.

⚠️ Wer das **Schreiben nach Exchange** einschaltet **und** zusätzlich den ICS-Feed abonniert, sieht jeden Termin zweimal. Beide Wege sind Absicht; man entscheidet sich für einen.

Einrichtung (für Administratoren)

Drei Schritte in der Domäne, danach ein Eintrag in der Plattform:

1. **Eigenes Dienstkonto** in Active Directory anlegen, z. B. `svc-ordivis-ews`, ohne interaktive Anmeldung. ⛔ **Nicht das LDAP-Abgleichskonto wiederverwenden** – das ist ein Konto zum Lesen des Verzeichnisses. Ein Konto, das plötzlich Kalender lesen kann, weil es zufällig schon da war, ist eine stille Rechteausweitung.
2. In Exchange die Rolle `ApplicationImpersonation` zuweisen, eingegrenzt über einen Verwaltungsbereich (`New-ManagementScope` / `New-ManagementRoleAssignment`). Die Rolle ist grob – sie erlaubt technisch jedes Postfach im Bereich. Begrenzt wird sie durch den Bereich **und** dadurch, dass die Plattform nur selbst freigegebene Postfächer öffnet.
3. Unter *Einstellungen → Meine Arbeit → Exchange* die **EWS-Adresse** eintragen (`https://exchange.example.local/EWS/Exchange.asmx`), Anmeldeart wählen und den **Verbindungstest** laufen lassen.

Anmeldeart. Läuft der Dienst unter einem Domänenkonto, meldet sich der HTTP-Kanal per Kerberos an – **ohne gespeichertes Kennwort**. Nur wenn der Server nicht domänengebunden ist, bleiben *hinterlegte Zugangsdaten*; das Kennwort wird dann verschlüsselt abgelegt und nie wieder angezeigt.

⚠️ **Exchange immer über den FQDN ansprechen, nie über die IP-Adresse.** Eine IP hat keinen Dienstprinzipalnamen; ohne SPN fällt die Anmeldung stillschweigend auf NTLM zurück. Die Plattform weist IP-Adressen deshalb schon beim Speichern ab, statt Sie später rätseln zu lassen.

TLS mit eigener Zertifizierungsstelle. Neben *Prüfen* (Vorgabe) gibt es *Eigene Zertifizierungsstelle* – Kettenfehler werden akzeptiert, **wenn** ein Glied der Kette den hinterlegten **Fingerabdruck** trägt; ein Namensfehler nie. Ohne Fingerabdruck lehnt die Plattform das Speichern ab, weil die Einstellung sonst nur ein anderer Name für „jedes Zertifikat akzeptieren" wäre.

Betrieb. Nach einem Fehlschlag wird mit wachsendem Abstand zurückgetreten (1, 2, 4 … Minuten, gedeckelt auf eine Stunde); nach fünf Fehlschlägen in Folge legt sich die Anbindung still. Wer gegen eine Drosselung endlos weiterläuft, wird nicht schneller, sondern gesperrt. Letzter Erfolg, letzter Fehler und Rückzugsfrist stehen in der Verwaltungsmaske, dort wird auch wieder freigegeben.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

10 Unterschiede zwischen Client und Portal

	Client	Portal
Ziehen und Ablegen	ja	**nein** – Verschieben über ein Menü je Karte
Kalenderansichten	Monat, Woche, Tag, Agenda	Monat, Woche, Agenda
Spalten bearbeiten	ja	nein
Team-Board, Übergabe	ja	nein

Dass das Portal nicht ziehen lässt, ist eine Entscheidung, keine Lücke: Ein Menü ist mit der Tastatur, auf dem Telefon und mit einem Screenreader bedienbar. Für ein Portal, dessen Zweck der gelegentliche Zugriff ist, ist das der bessere Tausch.

11 Rollen und Rechte

Recht	Erlaubt
`selforg.task.self`	Eigene Aufgaben, eigenes Brett, eigener Kalender, eigener Feed
`selforg.task.assign`	Aufgaben an andere übergeben (anbieten)
`selforg.board.team.read`	Team-Board ansehen
`selforg.board.team.manage`	Team-Boards anlegen und ihre Quelle festlegen

Die Exchange-**Anbindung** selbst wird unter `system.config.manage` verwaltet – die **Freigabe des eigenen Postfachs** dagegen nicht: Die trifft jeder für sich, mit `selforg.task.self`.

12 Was dieses Modul ausdrücklich nicht tut

- **Kein Ticketersatz.** Keine SLA, kein Melder, keine Eskalation, keine Warteschlange. Was einen Anspruch gegenüber Dritten begründet, gehört ins Ticketsystem.
- **Keine Zeiterfassung.** Der geschätzte Aufwand ist eine Schätzung; die tatsächliche Zeit wird am Ticket erfasst.
- **Keine Spiegelung von Exchange.** Es werden keine Exchange-Termine gespeichert; gelesen wird ein Zeitfenster, wenn jemand hinsieht.
- **Keine Einsicht für Vorgesetzte in persönliche Aufgaben.** Sichtbar wird nur, was über das Team-Board ausdrücklich freigegeben ist – und auch dort ohne Notizen.

Verwandte Themen

- **ID-11 – Agentenhandbuch – Service Desk** – Aufgaben aus der Ticketarbeit
- **ID-26 – Modul Workflows & Aufgaben** – Aufgaben aus Workflows
- **ID-20 – Modul Service Desk / Ticketing** – Termine am Ticket

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-31 – Modul Benachrichtigungen & E-Mail** – Erinnerungen und Glocke

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · DATENSICHERUNG

Modul Datensicherung

Sicherungen planen, ihre Ausführung protokollieren und die Kontrolle nachweisen

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-35 Datensicherung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Datensicherung

Die meisten Häuser sichern zuverlässig. Was fast keines beantworten kann, ist die Frage danach: *Wurde in den letzten dreißig Nächten tatsächlich gesichert – und wer hat das geprüft?* Die Antwort liegt üblicherweise in einem Postfach voller Berichtsmails, die niemand einzeln liest, und im Gedächtnis eines Kollegen.

Das Modul **Datensicherung** schließt genau diese Lücke.

Ordivis Platform sichert nicht selbst. Es mountet nichts, spricht kein Bandlaufwerk an und schreibt in kein S3-Bucket. Es ist das Führungssystem über Ihren vorhandenen Sicherungsprodukten – Veeam, Veritas, den Bordmitteln Ihres Hypervisors oder einem Skript. Diese Produkte sichern weiterhin; Ordivis Platform weiß, was sie tun sollen, hält fest, was sie getan haben, und führt die Kontrolle darüber.

Achtung – zwei Dinge heißen „Sicherung". Dieses Kapitel beschreibt die Sicherungen **Ihres Hauses**. Die Sicherung der Ordivis-Datenbank selbst ist eine Betriebsfunktion des Produkts und steht im Betriebshandbuch (ID-50).

INHALT

1 Der gefährlichste Lauf ist nicht der fehlgeschlagene
2 Die Bausteine
3 Einen Plan einrichten
4 Woher die Ergebnisse kommen
5 Die Kontrolle – und der Ein-Klick-Abschluss
5.1 Die geführte manuelle Prüfung: „Alle offenen als erfolgreich"
6 Aus einem Fehllauf einen Vorgang machen
7 Wiederherstellungstests
8 Die Regel 3-2-1-1-0
9 Bänder und Wechselmedien
10 Überblick, Kapazität und Konzept
11 Was Sie beim Umstieg erwartet
12 Rechte
13 Benachrichtigungen
Verwandte Themen

1 Der gefährlichste Lauf ist nicht der fehlgeschlagene

Ein fehlgeschlagener Sicherungslauf meldet sich. Er steht rot in der Konsole, er verschickt eine Mail, jemand sieht ihn.

Ein Sicherungsdienst, der seit elf Tagen gar nicht mehr startet, meldet **nichts**. Keine Fehlermeldung, keine Mail, keine Zeile. In jedem Werkzeug, das nur eingehende Meldungen sammelt, sieht dieser Zustand exakt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

so aus wie „keine Probleme".

Deshalb kennt Ordivis Platform nicht nur den *gemeldeten*, sondern auch den **erwarteten** Lauf. Aus dem hinterlegten Zeitplan entsteht für jede Nacht ein Eintrag. Trifft bis zum Ende des Toleranzfensters keine Meldung ein, bekommt dieser Eintrag den Zustand **ausgeblieben** – und der lässt sich filtern, zählen, melden und quittieren. Ein fehlender Datensatz könnte das nicht.

Genau dafür ist der Zeitplan ein Baukasten und kein Textfeld: Aus „täglich 22:00, So. Vollsicherung" lässt sich keine Erwartung ableiten.

2 Die Bausteine

Baustein	Was er beschreibt
Sicherungsplan	Die Zusage: Schutzklasse, Verantwortliche und Vertretung, Soll-RPO und Soll-RTO, Prüfrhythmus, gesetzliche Aufbewahrung
Sicherungsauftrag	Was das Produkt tatsächlich tut: Name im Produkt, ausführendes Backup-System, Verfahren, Zeitplan
Quelle	Welche Configuration Items der Auftrag sichert
Stufe	Ein Glied der Kopienkette – mit eigenem Ziel, eigenem Takt und eigener Aufbewahrung
Ziel	Wohin gesichert wird: Platte, SMB, NFS, S3, Band, Cloud-Dienst oder Replikat
Lauf	Ein Ergebnis je Auftrag, Stufe und Solltermin
Prüfperiode	Der Kontrollnachweis über einen Zeitraum
Wiederherstellungstest	Der einzige Nachweis, der etwas über Wiederherstellbarkeit aussagt

3 Einen Plan einrichten

💻 **Nur im Client:** *Datensicherung ▸ Pläne & Ziele* · 🔑 `backup.manage` — Das Portal führt nur die eigenen offenen Prüfungen (*Meine Vorgänge ▸ Datensicherung prüfen*); Pläne und Ziele werden im Client eingerichtet.

🔒 **Pflichtreihenfolge** — ein Plan braucht ein Ziel. Die vier Schritte unten sind nicht als Gliederung gemeint, sondern als Reihenfolge.

1. **Ziele zuerst.** Legen Sie an, wohin gesichert wird. Bei einem verschlüsselten Ziel ist der Nachweis der Schlüsselhinterlegung ein Pflichtfeld – als *Verweis* (Abschnitt im Notfallhandbuch, Tresorfach, Schlüsselverwaltung), niemals als Schlüssel selbst.
2. **Plan anlegen.** Name, Schutzklasse, Verantwortliche und Vertretung. Prüfrhythmus und Frist für den Wiederherstellungstest sind je Schutzklasse vorbelegt und bleiben änderbar.
3. **Auftrag anlegen.** Tragen Sie den Namen ein, unter dem Ihr Sicherungsprodukt den Auftrag führt – darüber ordnet Ordivis Platform später Berichtsmails und Schnittstellenmeldungen zu.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4. **Zeitplan bauen.** Wählen Sie die Wiederholung, die Startzeit, die Zeitzone, die Fensterlänge und die Toleranz. Die Schaltfläche **Nächste zehn Termine anzeigen** rechnet den Plan vor, bevor Sie speichern.
5. **Stufen anhängen.** Quelle → Platte → SMB → Band → S3, je Stufe ein Ziel und eine Aufbewahrung.

> **Tipp – die Terminvorschau ist mehr als eine Bequemlichkeit.** Sie zeigt auch, was über den Sommerzeitwechsel geschieht und was aus dem „31." in einem Februar wird (nämlich der Monatsletzte). Zwei Nächte im Jahr entscheiden sich hier zwischen einem richtigen und einem falschen Ausfallalarm.

4 Woher die Ergebnisse kommen

Drei Wege, die sich beliebig mischen lassen:

Weg	Wie er funktioniert	Wofür er taugt
Von Hand	Unter *Sicherungsläufe* → **Lauf erfassen**, auch rückwirkend	Funktioniert am ersten Tag und für jedes Produkt – auch für die Bandkassette, die jemand aus dem Tresor holt
Aus der Berichtsmail	Regeln je Auftrag: Absender, Betreffmuster, Erfolgs- und Fehlermuster	Das beste Verhältnis von Nutzen zu Aufwand – praktisch jedes Produkt kann eine Berichtsmail schicken
Über die Schnittstelle	`POST /api/v1/backup/runs` als Stapel, idempotent über die Lauf-Kennung der Quelle	Der sauberste Weg, setzt aber ein Skript bei Ihnen voraus

Trifft eine Meldung im Toleranzfenster eines erwarteten Laufs ein, wird **dieser** nachgetragen. So entsteht kein zweiter Eintrag, und derselbe Zeitraum zeigt nicht gleichzeitig einen erfolgreichen und einen ausgebliebenen Lauf.

> **Wichtig – eine Berichtsmail, die zu keiner Regel passt, verschwindet nicht.** Sie landet sichtbar in einem Eingang unter *Sicherungsläufe*. Ein geändertes Betreffformat nach einem Produktupdate fällt damit auf, statt still Läufe ausbleiben zu lassen.

5 Die Kontrolle – und der Ein-Klick-Abschluss

Prüfperioden entstehen **ohne Ihr Zutun** aus dem Rhythmus des Plans, mit Frist und Zuständigem. Sie finden sie unter **Datensicherung** → **Prüfungen**, im Portal unter *Datensicherung prüfen*.

Ist der Zeitraum lückenlos grün, genügt ein Klick auf **Zeitraum ohne Befund abschließen**.

Ist er es nicht, bleibt die Schaltfläche gesperrt – und nennt den Grund daneben. Dann führt der Weg über einen **Befund** je betroffenem Lauf: Ursache, Maßnahme, Stand. Erst wenn jeder nicht-grüne Lauf einen Befund trägt, lässt sich der Zeitraum abschließen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Warum die Sperre, wenn es doch einfach sein soll?** Beides gehört zusammen. Der Klick erledigt den Regelfall in zwei Sekunden – das ist der Grund, warum eine Kontrolle überhaupt durchgehalten wird. Eine Schaltfläche, die aber auch dann grün färbt, wenn drei Nächte fehlen, dokumentiert nicht die Kontrolle, sondern ihr Ausbleiben. Im Audit wäre sie **schlechter als gar nichts**, weil sie eine Prüfung behauptet, die nicht stattgefunden hat. Im Regelfall nimmt Ihnen die Bedingung nichts weg: Es ist alles grün, und der Klick sitzt.

Beim Abschluss wird **kein Haken** gespeichert, sondern ein **Nachweis**: Anzahl und Kennungen der Läufe, das gesicherte Volumen, Zeitpunkt und Prüfer. Ein Haken sagt „jemand hat geklickt". Der Nachweis sagt „am 03.08. bestätigte Frau X 31 von 31 erwarteten Läufen, 4,1 TB, kein Befund" – und *das* ist der Beleg nach ISO/IEC 27001 A.8.13. Eine spätere Korrektur legt eine neue Fassung daneben, statt die alte zu überschreiben.

5.1 Die geführte manuelle Prüfung: „Alle offenen als erfolgreich"

Nicht jedes Sicherungsprodukt ist angebunden. Wer die Ergebnisse **im Produkt nachsieht** und sie hier nachträgt, brauchte dafür bisher einen Lauf nach dem anderen. Dafür gibt es die Schaltfläche **Alle offenen als erfolgreich**: Sie trägt für alle noch offenen Läufe des gewählten Zeitraums ein erfolgreiches Ergebnis nach. Danach bleiben nur die Abweichungen zu bearbeiten.

Die Rückfrage nennt die Anzahl und ist bewusst deutlich formuliert:

> Das ist eine **Aussage über die Wirklichkeit** – bitte nur bestätigen, wenn im Sicherungsprodukt tatsächlich nachgesehen wurde. Eine Schaltfläche, die das lautlos täte, wäre dasselbe Problem wie ein Abschluss ohne Prüfung: Sie dokumentierte eine Kontrolle, die nicht stattgefunden hat.

Der Vorgang ist **wiederholbar**: Ein zweiter Klick trägt dieselben Läufe nicht erneut ein, sondern aktualisiert sie. Läufe, die bereits ein Ergebnis tragen — auch ein schlechtes —, bleiben unberührt; angefasst wird nur, was noch offen ist.

6 Aus einem Fehllauf einen Vorgang machen

> 💻 **Nur im Client:** *Datensicherung* ▸ *Sicherungsläufe* ▸ *Vorfall erzeugen* · 🔑 `backup.manage` und `tickets.update`

1. Öffnen Sie *Datensicherung* ▸ *Sicherungsläufe*.
2. Markieren Sie den beanstandeten Lauf.
3. Klicken Sie auf *Vorfall erzeugen*.

✅ **Ergebnis:** Es entsteht ein Vorgang im Service Desk – mit dem betroffenen Configuration Item verknüpft und dem Backup-Verantwortlichen des Plans zugewiesen.

Der Betreff entsteht automatisch:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
[Backup] VEEAM-SRV01 · Dateiserver FS-01 · 09.08.2026 22:00 · Fehlgeschlagen: VSS-Snapshot-Zeitüberschreitung
```

Die vollständige Meldung des Produkts steht im Beschreibungstext, das betroffene Configuration Item ist verknüpft, und zugewiesen wird der Backup-Verantwortliche des Plans – ersatzweise die Vertretung, ersatzweise die Support-Gruppe. Steht keiner davon fest, entsteht der Vorgang trotzdem: unzugewiesen ist besser als gar nicht.

Wichtig. Vierzehn Fehlnächte hintereinander erzeugen **einen** Vorgang mit vierzehn Journaleinträgen, nicht vierzehn Vorgänge. Ein geschlossener Vorgang wird bei erneutem Fehler innerhalb von vierzehn Tagen wieder aufgenommen.

7 Wiederherstellungstests

Ein Lauf mit dem Ergebnis „erfolgreich" bedeutet: *Das Sicherungsprodukt hat Erfolg gemeldet.* Er bedeutet **nicht**: *Die Daten sind wiederherstellbar.* Der Unterschied zwischen beidem ist der Wiederherstellungstest.

Erfasst werden Art (Einzeldatei, Postfach, Datenbank, vollständige VM, isolierte Testumgebung, Bare-Metal), Umfang, Ergebnis sowie die **gemessene** Wiederanlaufzeit und der **gemessene** Datenverlust.

- Ein überfälliger Test färbt die Ampel des Plans – **auch wenn jeder Lauf grün war**.
- Ist am Plan ein Wiederanlaufplan hinterlegt, erscheint der Test zugleich dort als Testergebnis (ID-30). Es gibt keinen zweiten Speicher.
- Eine Lücke zwischen gemessenem Ist und zugesagtem Soll wandert als Risiko in die Informationssicherheit (ID-28).
- Eine reine **Lesbarkeitsprüfung** eines Mediums erneuert die Testfrist bewusst nicht: Sie belegt, dass das Band lesbar ist, nicht dass sich daraus etwas wiederherstellen lässt.

8 Die Regel 3-2-1-1-0

Unter *Pläne & Ziele* zeigt **3-2-1-1-0** die Bewertung eines Plans. Sie wird **berechnet**, nicht angekreuzt – und nennt bei jeder nicht erfüllten Ziffer den Grund.

Ziffer	Woraus sie sich ergibt
3 Kopien	Produktivdaten plus Stufen mit gültiger Aufbewahrung. Ein Replikat zählt als Standort, nicht als Kopie
2 Medienarten	Verschiedene Zielarten über die Stufen hinweg
1 extern	Mindestens ein Ziel mit einem Standort, der nicht der Standort der Quelle ist. Ein Ziel ohne Standortangabe zählt nicht – unbekannt ist nicht extern
1 unveränderlich	Ein Ziel mit Object Lock, WORM oder Air-Gap

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ziffer	Woraus sie sich ergibt
0 Fehler	Kein fehlgeschlagener oder ausgebliebener Lauf im Zeitraum **und** ein bestandener Wiederherstellungstest innerhalb der Frist

Die letzte Ziffer ist die, die fast alle Werkzeuge weglassen – und die einzige, die etwas über Wiederherstellbarkeit aussagt.

9 Bänder und Wechselmedien

Für Häuser, die Band betreiben: Medienpools mit Rotationsverfahren, Einzelmedien mit Barcode, Zustand, Schreibzyklen gegen die Herstellergrenze und Ablauf der Aufbewahrung. Auslagerung und Rückholung werden mit **Quittung** festgehalten – wer hat das Band wann wohin gebracht, wer hat es angenommen.

> **Tipp.** Ohne diese Quittung ist die „1 extern" der 3-2-1-Regel eine Absicht. Mit ihr wird sie ein Datum, das jemand bestätigt hat.

Für ein Haus mit einer NAS und einer externen Platte ist dieser Abschnitt entbehrlich – Plan, Auftrag, Läufe und Kontrolle funktionieren ohne jede Medienverwaltung.

10 Überblick, Kapazität und Konzept

- **Überblick** zeigt je Plan eine Ampel samt Begründung, Fehler- und Ausfallquote, offene und überfällige Prüfungen sowie den Verlauf.
- **Auslastung der Sicherungsziele** schreibt das Wachstum fort und meldet, wann ein Ziel voll läuft – *bevor* es voll ist. Danach meldet es das Sicherungsprodukt ohnehin selbst, und dann ist die Nacht bereits verloren.
- Das **Datensicherungskonzept** erscheint als eigenes Kapitel im IT-Betriebshandbuch (ID-50). Es entsteht bei jedem Abruf aus den gepflegten Daten: Geltungsbereich, Verfahren, Ziele und Stufen, Fristen, Verantwortliche, Prüfrhythmus und Testnachweise.

11 Was Sie beim Umstieg erwartet

Bis zu dieser Fassung standen vier Angaben als Attribute am Configuration Item der Klasse *Backup-System*: Aufbewahrung, RPO, Zeitplan und der letzte Test-Restore. Sie ziehen beim Update automatisch in dieses Modul um und werden dort künftig geführt; doppelte Pflege entfällt.

> **Achtung – der übernommene Auftrag ist zunächst stillgelegt.** Aus dem bisherigen Zeitplan-Freitext lässt sich keine Erwartung ableiten. Wäre der Auftrag sofort aktiv, liefe er mit dem Vorgabetakt (täglich 22:00) los und Sie fänden am ersten Morgen eine Liste ausgebliebener Läufe vor, die niemand versprochen hat. Der Freitext bleibt vollständig in den Notizen des Auftrags erhalten: Bauen Sie den Zeitplan im Baukasten nach und schalten Sie den Auftrag dann scharf.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

12 Rechte

Recht	Wofür
`backup.read`	Pläne, Aufträge, Läufe und Nachweise sehen
`backup.manage`	Pläne, Aufträge, Stufen, Ziele und Einstellungen pflegen
`backup.run.record`	Ergebnisse von Läufen erfassen und nachtragen
`backup.verify`	Prüfperioden abschließen, Befunde und Wiederherstellungstests führen
`backup.media.manage`	Medienpools, Bänder und deren Auslagerung verwalten

Die Trennung folgt der Aufgabe, nicht dem Schema: Wer Läufe *erfasst* – etwa der Kollege, der das Band wechselt –, muss deshalb keine Pläne ändern dürfen. Und wer *prüft*, ist ein anderer als der, der die Sicherungsstrategie festlegt. Genau diese Trennung von Ausführung und Kontrolle will ein Prüfer sehen.

13 Benachrichtigungen

Sechs Ereignisse lassen sich je Benutzer für Glocke und E-Mail getrennt schalten (ID-31):

Ereignis	E-Mail voreingestellt	Warum
Sicherungslauf fehlgeschlagen	**an**	Verlangt eine Reaktion außerhalb der Anwendung
Sicherungslauf ausgeblieben	**an**	Der stille Fall – eine reine Meldung in der Anwendung bliebe wirkungslos
Prüfung fällig	aus	Routine
Prüfung überfällig	**an**	Eine ausgefallene Kontrolle ist der Auditbefund
Wiederherstellungstest fällig	aus	Planbar, mit Vorlauf
Ziel läuft voll	aus	Kapazitätswarnung mit Vorlauf

Verwandte Themen

- **ID-21 Modul CMDB** – die gesicherten Configuration Items und das Backup-System als CI
- **ID-30 Modul Notfall- & Wiederanlaufplanung (DR)** – dorthin schreiben die Wiederherstellungstests
- **ID-28 Modul Security (ISMS)** – Control A.8.13 und die RPO-/RTO-Lücke als Risiko
- **ID-29 Modul Business Continuity Management** – RPO und RTO aus der Business-Impact-Analyse
- **ID-20 Modul Service Desk** – die Vorgänge, die aus Fehlläufen entstehen
- **ID-31 Modul Benachrichtigungen & E-Mail-Integration** – Meldungen und die Aufnahme aus Berichtsmails

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-50 Betriebshandbuch** – das erzeugte Datensicherungskonzept und die Sicherung der Ordivis-Datenbank selbst
- **ID-44 Rollen- und Rechte-Matrix** – die fünf Rechte im Gesamtbild

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL V · ADMINISTRATION

Administrationshandbuch

Ersteinrichtung, Einstellungen, Jobs, Branding, Import/Export

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-40 Administration

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Administrationshandbuch

Dieses Dokument bündelt die zentralen Verwaltungsaufgaben. Spezialthemen sind ausgelagert: Benutzer & Rollen (ID-41), Mandantenfähigkeit (ID-42), Produktlizenzierung (ID-43), Installation/Serverbetrieb (ID-50). Alle Einstellungen erreichen Sie über **Einstellungen** im Client (Recht `system.config.manage`).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

1 Ersteinrichtung (First-Run)
2 Verzeichnisanbindung (AD/LDAP, SSO)
3 Service-Desk-Konfiguration
4 E-Mail-Integration
4.1 Fehlerberichte an den Hersteller-Support
4.2 Push-Benachrichtigungen an die Android-App
5 Geplante Aufgaben (Job-Scheduler)
5.1 Aufgaben und Mandanten
6 CI-Klassen-Verwaltung
7 Erscheinungsbild / Branding
7.1 Sprache der Oberfläche
8 Import / Export
8.1 Was sich per Datei importieren lässt
8.2 Die Vorlage ist der kürzeste Weg
8.3 Regeln für die Datei
8.4 Erst der Probelauf, dann der Bestand
9 Updates
9.1 Erkennung
9.2 Ablauf eines Update-Laufs
9.3 Was das Update sicher und verifizierbar macht
9.4 Update-Center
9.4.1 Arbeitsplatz-Client aktualisieren
9.5 Fortschritt und geprüftes Ergebnis
9.6 Update-Protokoll weitergeben
9.7 Rollendes Support-Fenster
9.8 Aufräumen des Delta-Stores (Retention)
10 Audit-Log
11 Modulverwaltung (Funktionsmodule ein-/ausschalten)
12 Plattform-Überwachung
13 Sicherheitsereignisse im Windows-Ereignisprotokoll
13.1 Ereignis-Referenz für Ihr SIEM
13.2 Kanal an ein SIEM weiterleiten
13.3 Warum diese Tabelle nicht von Hand gepflegt wird
14 Identitätsanbieter (SSO/OIDC)
15 Zugriffsüberprüfung (Access Review)
16 Mandantenübergreifende Freigaben
17 Automatisierungsregeln für Vorgänge
18 Protokollierung, Diagnose und Support-Paket
19 Datenschutz: Betroffenenrechte und AVV-Register
20 Eigenes Postfach für den Exchange-Kalender
21 Wichtige Rechte für Administration
Verwandte Themen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1 Ersteinrichtung (First-Run)

> 💻 **Nur im Client:** erscheint beim Erststart von selbst · 🔑 kein Recht nötig – es gibt noch keine Konten — Das Portal setzt einen eingerichteten Mandanten voraus und kann den Erststart nicht übernehmen.

🔒 **Pflichtreihenfolge** — der erste Administrator gehört zum Hauptmandanten. Ohne Schritt 1 gibt es niemanden, dem Schritt 2 ihn zuordnen könnte; der Assistent lässt die Schritte deshalb nicht tauschen.

1. Legen Sie den **Hauptmandanten** an (Name, Standort/Zentrale).
2. Legen Sie den **ersten Administrator** an – hier **mit** Kennwort. Das ist die einzige Stelle, an der ein Konto direkt mit Kennwort entsteht.
3. Bestätigen Sie. Ordivis Platform erzeugt automatisch die **Stammdaten**: Standardrollen, Standard-SLA-Profile und Standard-Kategorien.

✅ **Ergebnis:** Sie können sich mit dem angelegten Konto anmelden und danach weitere Mandanten, Benutzer und Konfigurationen ergänzen.

> **Achtung:** Das Kennwort des ersten Administrators ist der einzige Zugang zum frisch eingerichteten System. Es gibt an dieser Stelle **keinen** Wiederherstellungsweg – notieren Sie es sicher, bevor Sie fortfahren.

2 Verzeichnisanbindung (AD/LDAP, SSO)

- **AD/LDAP-Sync** konfigurieren (Server, Bind-DN, Basis-DN, Intervall) und **testen**.
- Der Sync ist **immer read-only** – Ordivis Platform schreibt nie ins AD.
- **SSO/SAML** koppelt externe Identitäten an Ordivis-Konten.

Abbildung 1 zeigt die Konfigurationsseite samt Verbindungstest.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

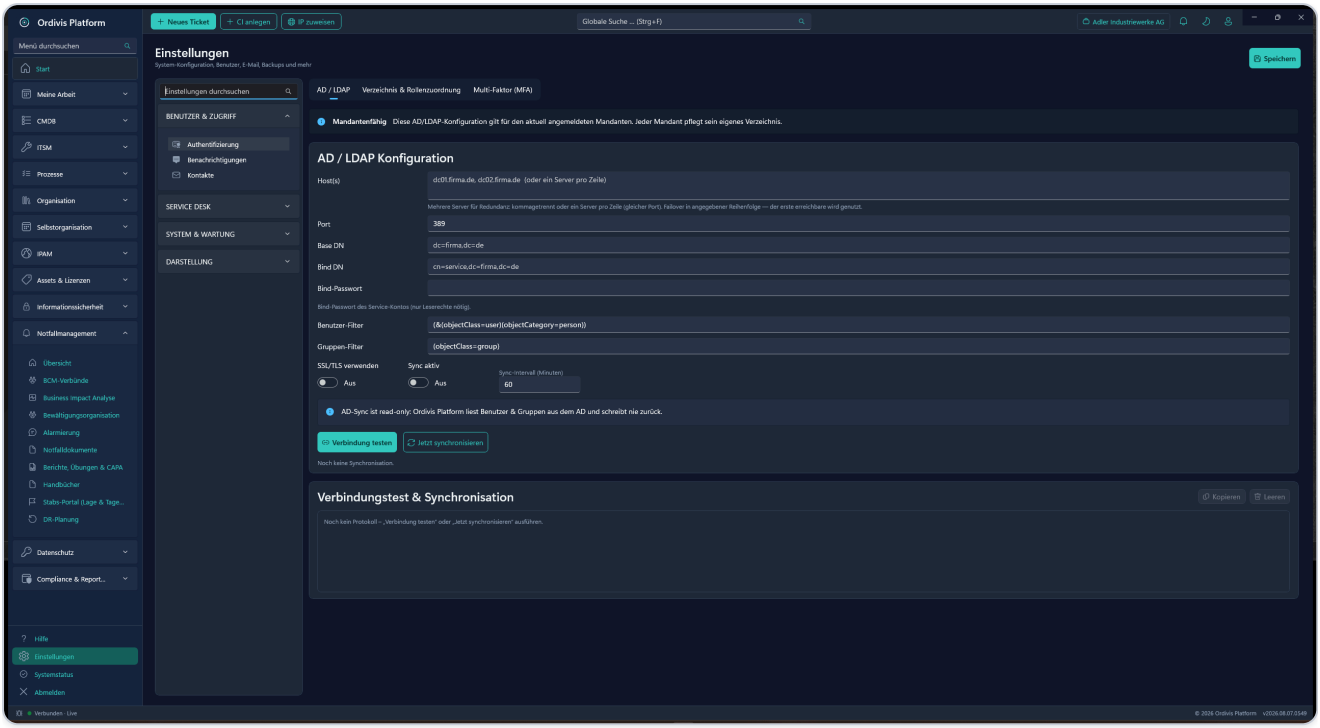

Abb. 1 – AD/LDAP-Konfiguration (mandantenspezifisch): Host, Base-/Bind-DN, Filter, Sync-Intervall und Verbindungstest – der Sync ist read-only.

Die Zuweisung von Rollen zu synchronisierten Benutzern erfolgt in der Benutzerverwaltung (ID-41).

3 Service-Desk-Konfiguration

Siehe ID-20, Abschnitt „Konfiguration“: SLA-Profile, Geschäftszeiten/Kalender, Kategorien + CMDB-Vorfilter, Auto-Routing, Freigabestufen, Wartungsfenster.

4 E-Mail-Integration

Siehe ID-31: SMTP-Versand, IMAP-Eingang, OAuth2, Benachrichtigungen, Präferenzen.

4.1 Fehlerberichte an den Hersteller-Support

Anwender können aus dem Client heraus einen Fehlerbericht senden (Käfer-Symbol in der Statusleiste, siehe ID-11). Für Sie als Administrator sind vier Dinge wichtig:

- **Der Versand läuft über das Outbound-Postfach Ihres Mandanten**, nicht über ein Postfach des Herstellers. Ohne eingerichteten E-Mail-Versand (ID-31) bleibt nur der lokale Weg: Der Client legt den Bericht dann als ZIP unter *Dokumente* ab und öffnet einen E-Mail-Entwurf.
- **Auf dem Server wird nichts gespeichert.** Der Bericht wandert in die Versand-Warteschlange und mit der Zustellung wieder heraus – es gibt keine Fehlerbericht-Tabelle.
- **Die Empfängeradresse ist konfigurierbar.** Voreingestellt ist `support@ordivis.eu`; über `Support:ReportAddress` in der `appsettings.json` leiten Sie die Berichte auf ein eigenes Postfach um, etwa wenn Ihr First Level sie zuerst sichten soll.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Der Bericht geht als von Hand geschriebene Mail hinaus** (`Auto-Submitted: no` nach RFC 3834) – anders als Benachrichtigungen und Eingangsbestätigungen, die als maschinell erzeugt gekennzeichnet sind. Nur so erzeugt ein Ticketsystem auf der Empfängerseite daraus einen Vorgang: Sein Schleifenschutz verwirft Maschinenmails, damit eine Abwesenheitsnotiz kein Ticket auslöst. Das ist besonders wichtig, wenn Sie mit `Support:ReportAddress` auf ein **eigenes Ticket-Postfach** umleiten. Als Antwortadresse trägt der Bericht die **meldende Person** – nicht das Postfach, über das er hinausging. Zur Zuordnung enthält er außerdem **Lizenznehmer** und **Mandantenname** (nicht dessen ID).

Der Endpunkt steht **jedem angemeldeten Benutzer** offen und verlangt bewusst kein eigenes Recht: Wer auf einen Fehler läuft, ist häufig der Anwender mit den wenigsten Rechten. Was mitgeschickt wird, entscheidet der Anwender im Dialog Punkt für Punkt; ohne Häkchen wird die betreffende Quelle gar nicht erst gelesen.

4.2 Push-Benachrichtigungen an die Android-App

Push ist Komfort, kein Zustellweg. Fällt die Zustellung aus, geht **keine Meldung verloren**: Sie bleibt in der Warteschlange, wird bis zu sechsmal mit wachsendem Abstand erneut versucht, und sie steht ohnehin in der Anwendung – die App zeigt sie beim nächsten Öffnen. Ein Ausfall des Zustellwegs ist damit ein Komfortverlust, kein Datenverlust.

Zwei Betriebsarten stehen zur Wahl (`Push:Mode` in der `appsettings.json`):

Wert	Bedeutung
`Relay` (Vorgabe)	Die Zustellung läuft über den Vermittler bei Grams IT. Auf diesem Server liegt **kein** Google-Schlüssel. Ausgewiesen wird sich mit dem Aktivierungs-Lease, das die Online-Aktivierung ohnehin führt.
`Direct`	Der Server sendet selbst an Firebase Cloud Messaging und braucht dafür einen eigenen Dienstkonto-Schlüssel im Abschnitt `Fcm` (DPAPI-geschützt). Nur sinnvoll mit einem **eigenen** Firebase-Projekt – das setzt einen eigenen App-Build voraus.

⚠️ **Ohne Online-Aktivierung gibt es kein Push.** Im Vermittler-Betrieb ist das Aktivierungs-Lease der Ausweis; eine Anlage ohne Verbindung ins Internet hat keines. Das ist keine Einschränkung des Vermittlers: Ohne Weg ins Internet erreicht auch der unmittelbare Versand Googles Zustelldienst nicht.

Nach einem Neustart steht das Lease erst mit dem ersten Lizenz-Heartbeat (Vorgabe: zehn Minuten). Meldungen aus diesem Fenster werden deshalb als **vorübergehend gescheitert** geführt und wiederholt – nicht verworfen.

Welche Daten dabei wohin fließen und was protokolliert wird, steht öffentlich unter https://ordivis.eu/unterauftragsverarbeiter.html.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5 Geplante Aufgaben (Job-Scheduler)

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Geplante Aufgaben* · 🔑 `system.config.manage` — Das Portal zeigt den Betriebszustand unter *Compliance & Reporting* ▸ *Betriebssicht*, steuert die Aufgaben aber nicht.

Der zentrale **Managed Job Scheduler** verwaltet wiederkehrende Aufgaben:

1. Öffnen Sie *Einstellungen* ▸ *Geplante Aufgaben*.
2. Sehen Sie die Liste durch – je Job stehen dort *Zeitplan*, *Status* und der letzte Lauf.
3. Schalten Sie einen Job über den Schalter in seiner Zeile **aktiv** oder **inaktiv**.
4. Lösen Sie einen Job bei Bedarf über *Jetzt ausführen* **manuell** aus.

✅ **Ergebnis:** Der Job läuft zum eingestellten Zeitpunkt. Nach einem manuellen Lauf steht das Ergebnis in der Spalte für den letzten Lauf.

> **Tipp:** Ein Job, der seit Tagen nicht mehr gelaufen ist, fällt in dieser Liste sofort auf – und sonst nirgends. Sehen Sie hier turnusmäßig nach.

Abbildung 2 zeigt die Liste der Aufgaben.

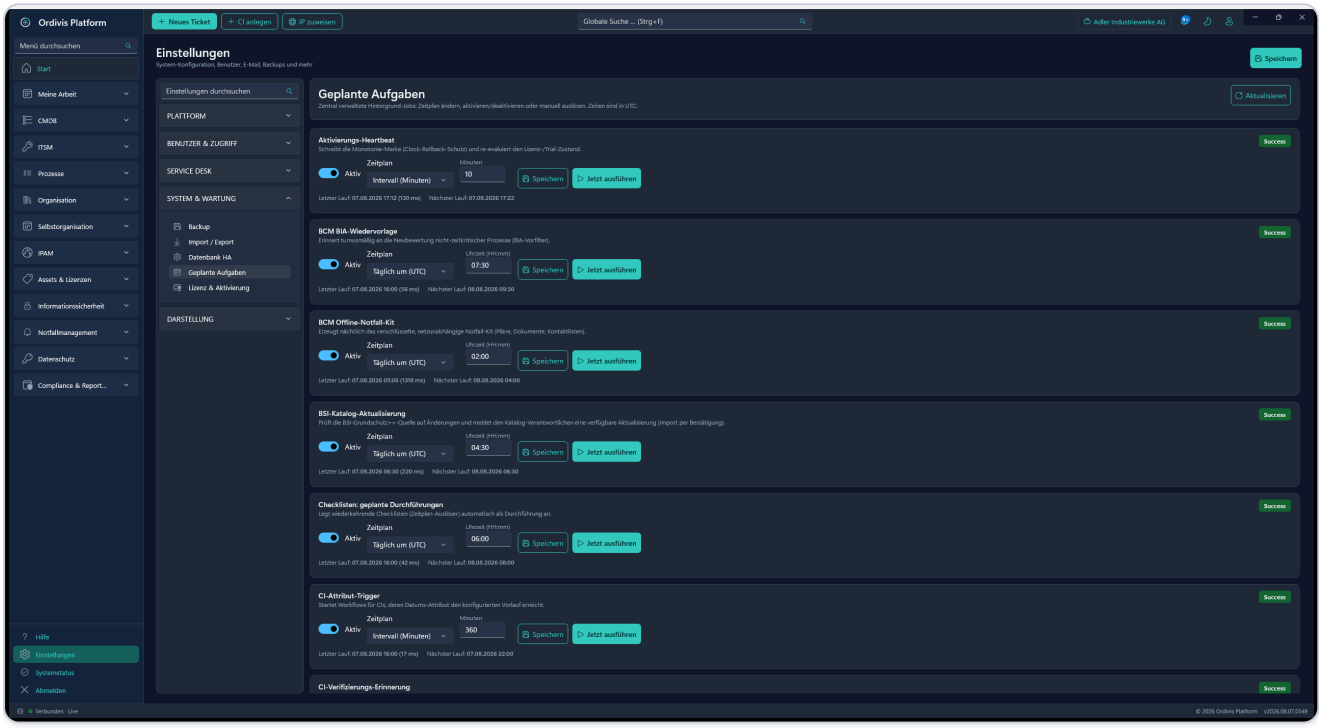

Abb. 2 – Geplante Aufgaben: zentral verwaltete Hintergrund-Jobs mit Zeitplan, letztem/nächstem Lauf und Status.

> **Hinweis:** Einige Integrationen (IMAP, LDAP, Backup, geplante Berichte) bringen eigene Zeitplan-Konfiguration mit und erscheinen zusätzlich in ihrem jeweiligen Bereich.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5.1 Aufgaben und Mandanten

Die Liste ist **installationsweit**: Jede Aufgabe steht genau einmal darin, und Zeitplan, Status und letzter Lauf gelten für die gesamte Installation – nicht je Mandant. Was eine Aufgabe *bearbeitet*, hängt dagegen von ihrer Art ab:

- **Mandantenbezogene Aufgaben** (der weit überwiegende Teil: SLA-Fristen, Wiedervorlagen, Vertrags- und Lizenzerinnerungen, Postfachabruf, Mailversand, Kennzahlen) laufen bei jedem Lauf **nacheinander für jeden aktiven Mandanten** und arbeiten dabei jeweils in dessen eigener Datenbank.
- **Plattformaufgaben** (Update-Prüfung, Lizenz-Heartbeat, Plattformüberwachung) betreffen die Installation als Ganzes und laufen genau einmal.

Daraus folgen zwei Dinge für den Betrieb: Der angezeigte Status **Erfolgreich** bedeutet, dass der Durchgang über *alle* Mandanten fehlerfrei war; und die Laufzeit einer mandantenbezogenen Aufgabe wächst mit der Zahl der Mandanten. Läuft eine Aufgabe noch, wird sie beim nächsten fälligen Termin nicht ein zweites Mal gestartet, sondern übersprungen – bei sehr kurzen Intervallen und vielen Mandanten ist deshalb ein größerer Abstand die bessere Einstellung.

6 CI-Klassen-Verwaltung

Abbildung 3 zeigt die konfigurierten Klassen mit der Spalte *Felder*.

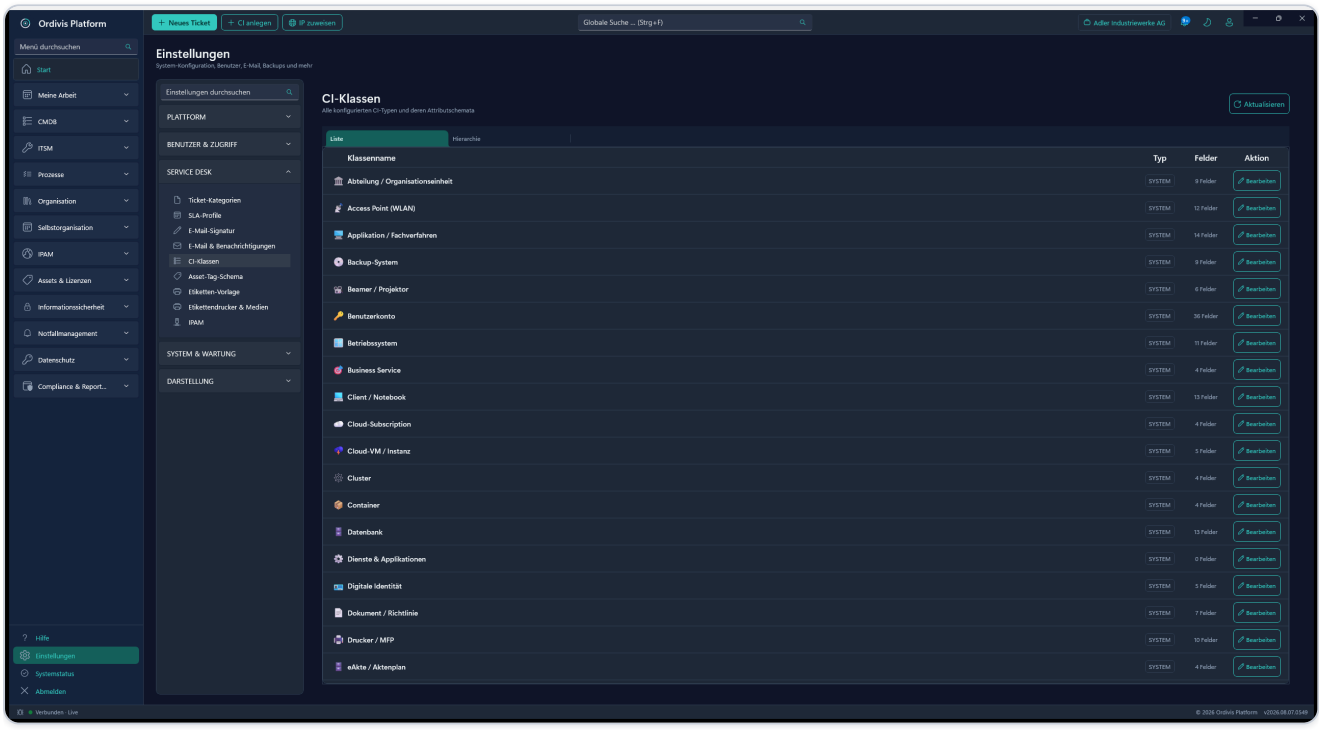

Abb. 3 – CI-Klassen: alle konfigurierten CI-Typen mit Attributschema (Spalte „Felder").

Unter **CI-Klassen** definieren Sie Klassen und Attribute der CMDB (ID-21). Neue Attribute werden zentral angelegt und erscheinen dann automatisch im CI-Formular.

Daneben stehen zwei Einstellungsseiten für die Kennzeichnung der Geräte (Einzelheiten in ID-21, Abschnitt „Etiketten und NFC-Tags am Gerät"):

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Etiketten-Vorlage** – was auf dem Etikett steht (Barcode-Art, Layout, sichtbare Felder, Logo). Genau eine Vorlage je Mandant.
- **Etikettendrucker & Medien** – worauf gedruckt wird: Maße, Auflösung und Sprache des Geräts (PDF-Bogen, Zebra ZPL, Brother, Epson). Davon dürfen mehrere nebeneinander bestehen; höchstens eines ist die Vorgabe. Ein Testdruck lässt sich erzeugen, bevor das Medium gespeichert wird.

7 Erscheinungsbild / Branding

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Erscheinungsbild* bzw. ▸ *Branding* · 🔑 kein Recht für das persönliche Erscheinungsbild, `system.config.manage` für das Branding — Das Portal bietet das persönliche Erscheinungsbild im Paletten-Menü an (ID-12), das mandantenweite Branding jedoch nicht.

Ordivis Platform trennt hier bewusst zwei Ebenen:

- **Erscheinungsbild** (pro Benutzer): heller/dunkler Modus und Farbschema (Material Design 3), dazu Barrierefreiheit (hoher Kontrast, Schriftgröße, reduzierte Bewegung, dauerhaft sichtbarer Fokus) und Sprache.
- **Branding** (pro Mandant): **Firmenname und Logo für generierte Dokumente**. Beide erscheinen als Kopf auf allen Reports und PDFs – Report-Exporte sowie ISMS-/BCM-Berichte (Risikoregister, Schutzbedarf, Erklärung zur Anwendbarkeit, Notfallplan u. a.). In der Fußzeile stehen weiterhin die Ordivis-Kennzeichnung („Erstellt durch Ordivis Platform“) und die Seitenzahl. Das Logo (PNG/JPEG, max. 2 MB) wird zentral hinterlegt und mandantenweise verwendet; Ändern erfordert `system.config.manage`.

Abbildung 4 zeigt die Einstellungen dazu.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

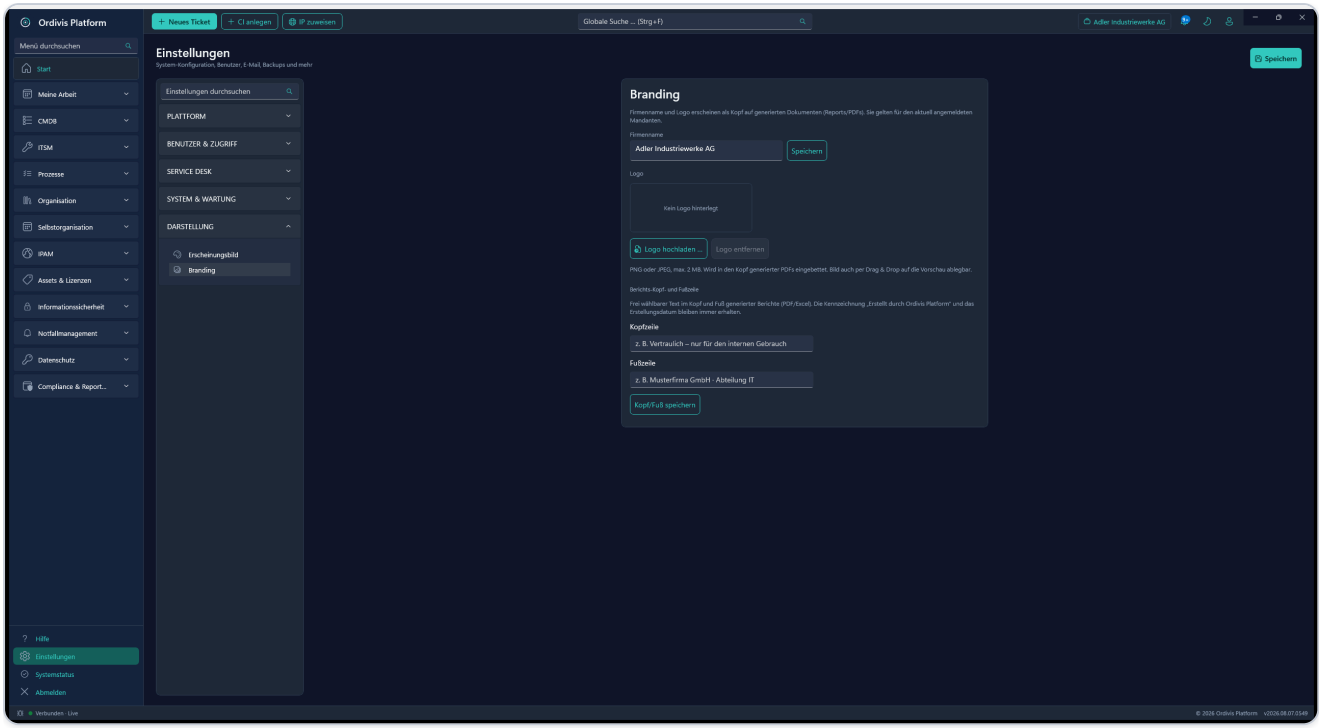

Abb. 4 – Branding (pro Mandant): Firmenname und Logo für den Kopf generierter Reports und PDFs; die Ordivis-Kennzeichnung bleibt in der Fußzeile.

7.1 Sprache der Oberfläche

> 💻 **Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Erscheinungsbild* ▸ *Sprache* · 🌐 **Portal:** Flaggensymbol in der Kopfzeile · 🔑 kein Recht nötig — die Sprache gilt je Benutzer, nicht je Mandant

Die Oberfläche liegt in **Deutsch, Englisch und Französisch** vor. Deutsch ist die Produktsprache; Englisch und Französisch sind vollständig übersetzt.

Im Client stehen Deutsch und Englisch als Schalter bereit, alle weiteren Sprachen in der Liste darunter. Die Namen stehen dort in der jeweiligen Sprache selbst — wer die Oberfläche gerade nicht lesen kann, findet *Français*, aber nicht *Französisch*.

Was Sie wissen sollten:

- Die Umschaltung wirkt im Client **nach einem Neustart des Programms**. Das Portal lädt die Seite selbst neu und ist sofort umgestellt.
- Ist eine Sprache nicht vollständig übersetzt, erscheinen die **fehlenden Texte auf Deutsch**. Es bleibt nichts leer und nichts bricht ab.
- Nicht übersetzt werden **Ihre eigenen Daten**: Ticketbetreffs, CI-Namen, Notizen und Dokumente stehen in der Sprache, in der sie erfasst wurden. Die Oberfläche wechselt, der Inhalt nicht.
- **Datums-, Zeit- und Zahlenformate** folgen der gewählten Sprache. Aus dem 15.08.2026 wird im Englischen 8/15/2026 — dieselbe Angabe, andere Schreibweise.
- Weitere Sprachen sind vorbereitet, aber noch nicht übersetzt; sie erscheinen erst in der Auswahl, wenn ihre Übersetzung vorliegt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8 Import / Export

Tabellen bieten überall **CSV-/PDF-Export**. Für den Weg hinein – den Dateiimport – gilt Folgendes.

8.1 Was sich per Datei importieren lässt

Bereich	Wo der Import läuft	Was entsteht
CMDB – Configuration Items	CI-Liste ▸ *Import*, oder Einstellungen ▸ Import / Export	CIs einer wählbaren CI-Klasse
Organisation – Aufbauorganisation	Organisation ▸ Aufbauorganisation ▸ *Import*	Einheiten, Stellen, Besetzungen, Rollen
Prozesse – Steckbriefe	Prozesse ▸ Prozessdokumentation ▸ *Import*	Prozesse samt Steckbriefangaben

Andere Bereiche – etwa Tickets oder Assets – haben **keinen** Dateiimport. Die Auswahl in den Einstellungen nennt deshalb genau die Bereiche, die einen haben; sie wird vom Server geliefert und kann nicht mit der Wirklichkeit auseinanderlaufen.

8.2 Die Vorlage ist der kürzeste Weg

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Import / Export* · 🔑 `cmdb.ci.update` — Das Portal hat keinen Dateiimport.

🔒 **Pflichtreihenfolge** — die Vorlage wird **je Bereich und CI-Klasse erzeugt**. Wer sie vor der Auswahl herunterlädt, bekommt die Spalten eines anderen Bereichs.

1. Öffnen Sie *Einstellungen* ▸ *Import / Export*.
2. Wählen Sie den **Bereich** – bei der CMDB zusätzlich die **CI-Klasse**.
3. Laden Sie die **Beispieldatei** als CSV oder XLSX herunter.
4. Überschreiben Sie die Beispielzeile mit Ihren Daten.
5. Laden Sie die ausgefüllte Datei hoch und prüfen Sie die **Vorschau**, bevor Sie übernehmen.

✅ **Ergebnis:** Der Import läuft gegen genau die Felder, die der Bereich kennt – ohne dass Sie die Kopfzeile erraten mussten.

Die Beispieldatei enthält:

- die **vollständige Kopfzeile** – alle möglichen Felder des gewählten Bereichs,
- eine **Beispielzeile** mit gültigen Werten, die Sie überschreiben,
- bei XLSX ein zweites Blatt **Feldbeschreibung** mit Typ, Pflichtkennzeichen, Bedeutung und den zulässigen Werten je Spalte.

Die Vorlage wird **erzeugt, nicht gepflegt**: Ihre Spalten kommen aus derselben Quelle, aus der der Import liest. Legen Sie ein neues Attribut an einer CI-Klasse an, steht es ohne weiteres Zutun in der Vorlage. Dieselbe Übersicht zeigt die Schaltfläche **Aufbau der Datei anzeigen** direkt in der Anwendung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Read-only-Felder fehlen bewusst.** Werte, die Ordivis Platform selbst berechnet (etwa aus einer aktiven Asset-Zuweisung), stehen nicht in der Vorlage – ein Wert in der Datei wäre eine Behauptung, die beim nächsten Neuberechnen verschwindet.

8.3 Regeln für die Datei

- **Kopfzeile:** genau die Spaltennamen der Vorlage. Reihenfolge und Groß-/Kleinschreibung sind egal. Nicht benötigte Spalten dürfen leer bleiben oder entfallen – außer den Pflichtspalten.
- **Trennzeichen:** Semikolon oder Komma; Ordivis Platform erkennt es an der Kopfzeile. Felder mit Trennzeichen gehören in Anführungszeichen.
- **Zeichensatz:** UTF-8. Die heruntergeladene CSV-Vorlage trägt eine Byte-Reihenfolge-Marke, damit Excel Umlaute im deutschen Gebietsschema richtig öffnet.
- **Datum:** `2026-12-31` oder `31.12.2026`.
- **Zahlen:** `1234.56` oder `1234,56`, **ohne Tausendertrennzeichen**. `1.234` wird als 1,234 gelesen, nicht als 1234.
- **Unbekannte Spalten** werden gemeldet, nicht stillschweigend übergangen – ein Tippfehler in der Kopfzeile fällt damit auf, statt eine ganze Spalte verschwinden zu lassen.

8.4 Erst der Probelauf, dann der Bestand

Jeder Import beginnt mit einem **Probelauf**: Er prüft jede Zeile, nennt die Zahl der Objekte, die entstehen würden, und listet die Beanstandungen mit Zeilennummer und Feld. Erst nach Ihrer ausdrücklichen Zustimmung wird geschrieben. Der Knopf zum Zustimmen ist bewusst **nicht** vorbelegt.

Bei CIs erkennt der Import Dubletten an der **Seriennummer**: Ist sie bereits vergeben, wird die Zeile übersprungen statt ein zweites Objekt anzulegen. Kommt dieselbe Seriennummer zweimal in *einer* Datei vor, wird die zweite Zeile beanstandet.

Für den Import gilt ein eigenes Recht (`cmdb.import` bei CIs, `gvp.manage` bzw. `process.manage` bei Organisation und Prozessen) – Leserechte allein genügen nicht.

9 Updates

Ordivis Platform aktualisiert sich als **Gesamtprodukt** (API, Web-Portal, WinUI-Client, Wandschirm) über **signaturgeprüfte Delta-Pakete** – übertragen werden nur die tatsächlich geänderten Dateien, nicht das vollständige Setup. Zentrale Anlaufstelle ist **Einstellungen ▸ Updates** im Client (Administratoren: Prüfen/Anzeigen mit `activation.manage`, Auslösen mit `system.config.manage`).

Dieses Kapitel beschreibt das Update-System aus Sicht des Administrators und – im Mittelpunkt – die **Mechanismen, die ein Update sicher und nachprüfbar machen** (Abschnitt 9.3). Die serverseitigen Betriebsdetails (Endpunkte, Verzeichnisse, Fehlerbehebung) stehen im Betriebshandbuch (ID-50).

9.1 Erkennung

Die API prüft serverseitig gegen die Website-`version.json` auf neue Versionen und informiert Administratoren (In-App / E-Mail / Portal), dedupliziert je Version (Ereignis `system.update.available`).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Die Prüfung läuft per Zeitplan-Job und lässt sich über **„Nach Updates suchen"** manuell anstoßen.

9.2 Ablauf eines Update-Laufs

Vom Auslösen bis zum geprüften Ergebnis durchläuft jedes Update dieselben Schritte:

1. **Manifest prüfen.** Die API lädt das signierte Release-Manifest der Zielversion und prüft **Signatur** und **Versionsbindung**. Schlägt das fehl, endet der Lauf sofort – es wird nichts verändert.
2. **Datenbank sichern.** Vor jedem Eingriff wird das tägliche DB-Backup angestoßen (Sicherheitsnetz, u. a. für Datenbank-Migrationen).
3. **Delta laden (Staging).** Das Produkt vergleicht die installierten Datei-Hashes mit dem Manifest und lädt **nur die abweichenden** Dateien – jede **einzeln SHA-256-geprüft** – in ein Staging-Verzeichnis. Die laufende Installation bleibt dabei unberührt (**nicht-destruktiv**).
4. **An den eigenständigen Updater übergeben.** Die API stellt einen **Gesamtplan** über alle betroffenen Komponenten zusammen und startet den **entkoppelten Updater**. Dieser läuft eigenständig weiter, auch wenn API oder Client währenddessen beendet werden.
5. **Komponenten in Reihenfolge übernehmen.** Der Updater arbeitet den Plan ab – **Dienste zuerst** (Portal, dann API), den **Client zuletzt**. Je Komponente: Dienst **sauber stoppen** → Dateien mit **Snapshot** tauschen → Dienst **starten** → **Health-Check**. Bei Fehler: **Rollback** auf die Vorversion.
6. **Client zuletzt neu starten.** Wurde das Update manuell am Client ausgelöst, startet der Updater den Client **erst dann** neu, wenn die API wieder gesund ist. Ein reines Server-/Auto-Update lässt einen bewusst geschlossenen Client geschlossen.
7. **Geprüftes Ergebnis ablegen.** Der Updater schreibt das Ergebnis je Komponente nach `update-status.json`; der Client zeigt es beim nächsten Start an (Abschnitt 9.5).

> **Tipp:** Weil die Ausführung entkoppelt ist, meldet das Fortschrittsfenster nach dem Übergeben „**an den Updater übergeben – wird ausgeführt und geprüft**". Das ist bewusst **kein** vorschnelles „fertig"; der endgültige, geprüfte Ausgang erscheint nach dem Neustart unter *Einstellungen* ▸ *Updates*.

9.3 Was das Update sicher und verifizierbar macht

Ein Update greift tief in eine laufende Installation ein. Ordivis Platform behandelt es deshalb als **sicherheitskritische Operation**: Mehrere voneinander unabhängige Mechanismen greifen ineinander, und **keiner verlässt sich darauf, dass ein anderer funktioniert hat**. Sie beantworten drei Fragen.

Kommt das Update wirklich vom Hersteller – und ist es unverändert? (Authentizität & Integrität)

- **Signiertes Release-Manifest.** Zu jeder Version gehört ein `manifest.json`, das je Komponente **jede** Datei mit ihrem **SHA-256-Fingerabdruck** und ihrer Größe auflistet. Das Manifest ist mit **ECDSA / ES256** (Kurve P-256, kompaktes JWS) **signiert**. Der **private** Schlüssel liegt ausschließlich beim Hersteller; der **öffentliche** ist fest in das Produkt eingebaut. Ein Manifest ohne gültige Signatur wird **nie** angewandt – der Lauf beginnt gar nicht erst.
- **Bindung an die Zielversion.** Ein zwar korrekt signiertes, aber für eine *andere* Version ausgestelltes Manifest wird abgelehnt. Ein altes (womöglich verwundbares) oder verwechseltes Paket lässt sich damit nicht unterschieben.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Jede Datei einzeln geprüft.** Die Dateien liegen **inhaltsadressiert** ab – sie werden über ihren Hash angefordert und nach dem Laden **erneut gehasht** und gegen den signierten Manifest-Wert geprüft. Weicht ein Hash ab, wird die Datei verworfen und der Lauf abgebrochen – **auch dann, wenn der Download- oder Spiegelserver manipuliert wäre**.
- **Zwei Schutzschichten.** Die **Signatur** beweist, dass die *Datei-Liste* echt ist; der **Hash je Datei** beweist, dass *jede geladene Datei* exakt dieser Liste entspricht. Ein manipuliertes Netzwerk oder CDN kann so keinen fremden Code einschleusen.
- **Rechtetrennung.** Nur die Plattform-Ebene darf ein Update **auslösen** (`system.config.manage`); schon das **Anzeigen** des Umfangs erfordert `activation.manage` .

Wird nur ein geprüfter Zustand übernommen – und lässt er sich zurücknehmen? (Verifizierbarkeit & Reversibilität)

- **Snapshot + automatischer Rollback je Komponente.** Vor dem Dateitausch wird ein **Snapshot** der Installation angelegt. Scheitert *irgendein* Schritt – Kopieren, Dienststart oder Health-Prüfung –, wird der Snapshot **zurückgespielt** und die **Vorversion** gestartet. Einen „halb aktualisierten" Stand gibt es nicht.
- **Erfolg wird gemessen, nicht angenommen.** Eine Dienst-Komponente gilt erst dann als aktualisiert, wenn ihr Dienst wieder **läuft** *und* ihr **Health-Endpunkt** „gesund" meldet. Andernfalls: Rollback.
- **Geschützter Laufzeit- und Konfigurationsstand.** `appsettings*` , Schlüsseldateien (`*.key`) und der `data/` -Ordner (u. a. der Schlüssel des Mail-Tresors) werden **nie** angefasst – weder beim Entfernen überzähliger Dateien noch beim Rollback. Das verhindert die frühere Fehlerklasse, bei der ein Update Konfiguration oder gespeicherte Geheimnisse verlieren konnte.
- **Entkoppelte Ausführung.** Die Übernahme läuft in einem **eigenständigen Updater-Prozess**, der den Stopp von API und Client **überlebt**. Weil kein laufender Prozess seine eigenen Dateien tauscht, entstehen **keine Datei-Sperren** und damit **keine Teil-Updates**.
- **Feste Reihenfolge, Client zuletzt.** Erst die Dienste (Portal, dann API), dann der Client – dieser kommt **erst hoch, wenn die API aktualisiert und gesund ist**.
- **Sicherheitsnetz vor dem Eingriff.** Vor dem Laden der Dateien wird das **Datenbank-Backup** angestoßen; das Staging selbst ist **nicht-destruktiv**.

Kann man hinterher zweifelsfrei sehen, was passiert ist? (Nachvollziehbarkeit)

- **Geprüftes Ergebnis je Komponente.** Der Updater schreibt Fortschritt und geprüftes Endergebnis in `update-status.json` . Der Client zeigt nach dem Neustart **genau dieses** Ergebnis – Erfolg, Rollback oder Fehler je Komponente (Abschnitt 9.5).
- **Revisionssicheres Audit-Ereignis.** Jeder Lauf erzeugt `system.update.performed` mit Zielversion und Ergebnis (Abschnitt 10).
- **Detailprotokoll je Lauf.** Host- und Updater-Protokolle unter `%ProgramData%\InfraDesk\update-logs\` halten jeden Schritt mit Zeitstempel fest; im Update-Fenster ist der Text per Klick kopierbar (Abschnitt 9.6).

Kurzüberblick der Mechanismen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Mechanismus	Sichert …	Woran erkennbar / prüfbar
Signiertes Manifest (ES256)	Echtheit – kommt vom Hersteller	ungültige Signatur → Lauf startet nicht
Versionsbindung des Manifests	kein Unterschieben fremder/alter Pakete	Fremdversion → „passt nicht zur angeforderten"
SHA-256 je Datei (inhaltsadressiert)	Unversehrtheit jeder Datei – auch bei manipuliertem Server	Hash-Abweichung → Datei verworfen, Abbruch
Snapshot + Rollback	kein halb aktualisierter Stand	Fehler → Vorversion läuft wieder
Health-Check	nur ein lauffähiger Zustand zählt als Erfolg	„gesund"-Prüfung nach Start, sonst Rollback
Geschützter Laufzeitstand	Erhalt von Konfiguration/Geheimnissen	`appsettings*` , `*.key` , `data/` bleiben unberührt
Entkoppelter Updater	keine Datei-Sperren, keine Teil-Updates	Ausführung überlebt Stopp von API/Client
Pre-Update-DB-Backup	Rückfallpunkt bei Datenfehlern	Backup vor dem Lauf angestoßen
`update-status.json`	geprüftes Ergebnis statt Vermutung	Client zeigt Ausgang je Komponente
Audit + Update-Protokoll	lückenlose Nachvollziehbarkeit	`system.update.performed` , `update-logs\`

Wichtig: Diese Prüfungen sind fest eingebaut und lassen sich nicht abschalten. Erst wenn die **Signatur und alle** Datei-Hashes stimmen, wird überhaupt etwas übernommen; andernfalls bleibt die vorhandene Installation unverändert.

9.4 Update-Center

💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Updates* · 🔑 `system.config.manage` — Das Portal wird vom Server-Update mit aktualisiert, hat aber keine eigene Update-Seite.

Administratoren finden hier alles an einer Stelle:

- **Auto-Update-Policy in drei Stufen**: **Aus** / **Nur benachrichtigen** (erkennen + melden) / **Vollautomatisch** – bei Vollautomatik wendet der Server das geprüfte Delta selbsttätig im konfigurierten **Wartungsfenster** (Wochentage + Uhrzeit) an (Health + Rollback wie manuell). Die Policy wird gespeichert und steuert Server-Verhalten und Anzeige.
- **Integriertes Changelog**: die Website führt neben `version.json` eine fortgeschriebene `changelog.json`; das aufbereitete Änderungsprotokoll erscheint direkt im Client – kein Wechsel zur Website nötig. Einträge neuer als die installierte Version sind mit **„Neu in dieser Version"** markiert.
- **Komponentenansicht**: je Komponente (API / Portal / Client / Wallboard) installierte Version, verfügbare Zielversion und ob sie vom Update **betroffen** ist (aus dem signaturgeprüften Delta-Plan, inkl.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Dateizahl/MB).

- **Manuelles Update**: eigener Ein-Klick-Weg „Jetzt manuell aktualisieren" – prüft sofort und wendet ein gültig signiertes Delta direkt an, unabhängig von der eingestellten Auto-Update-Policy.

> **Diese Seite zeigt zwei Rechner.** Versionsinfo und Komponentenansicht beschreiben den **Server** – auch die Zeile „Client", die für die auf dem Server abgelegten Client-Dateien steht. Die Version des Arbeitsplatzes, an dem Sie gerade sitzen, steht in der **Statusleiste unten rechts**. Beide können auseinanderlaufen: Ein auf einem eigenen PC installierter Arbeitsplatz-Client wird vom Server-Update nicht erfasst.

9.4.1 Arbeitsplatz-Client aktualisieren

Liegt der Arbeitsplatz hinter dem Server, meldet die Statusanzeige des Update-Centers **„Server aktuell – dieser Arbeitsplatz veraltet"** und nennt beide Versionen. **„Jetzt aktualisieren"** aktualisiert dann **diesen Arbeitsplatz** über den mitgelieferten Aktualisierungsdienst: Der Client wird dafür beendet, die Dateien werden nach Signatur- und Prüfsummenkontrolle getauscht und der Client neu gestartet; bei einem Fehlschlag kehrt er zur Vorversion zurück. Die Dateien kommen über den eigenen Server – der Arbeitsplatz braucht **keinen Internetzugang**.

Derselbe Weg steht im Hinweisbanner am oberen Rand zur Verfügung. Weil die Installation üblicherweise unter *Programme* liegt, fragt Windows nach Administratorrechten; wird das abgelehnt, meldet Ordivis Platform das ausdrücklich. Fehlt der Aktualisierungsdienst – bei Installationen aus der Zeit vor 2026.07.21.2006 – wird stattdessen auf die **Setup-Datei** aus dem Download-Bereich verwiesen; danach greift das Selbstupdate.

Abbildung 5 zeigt das Update-Center mit einer verfügbaren Version.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

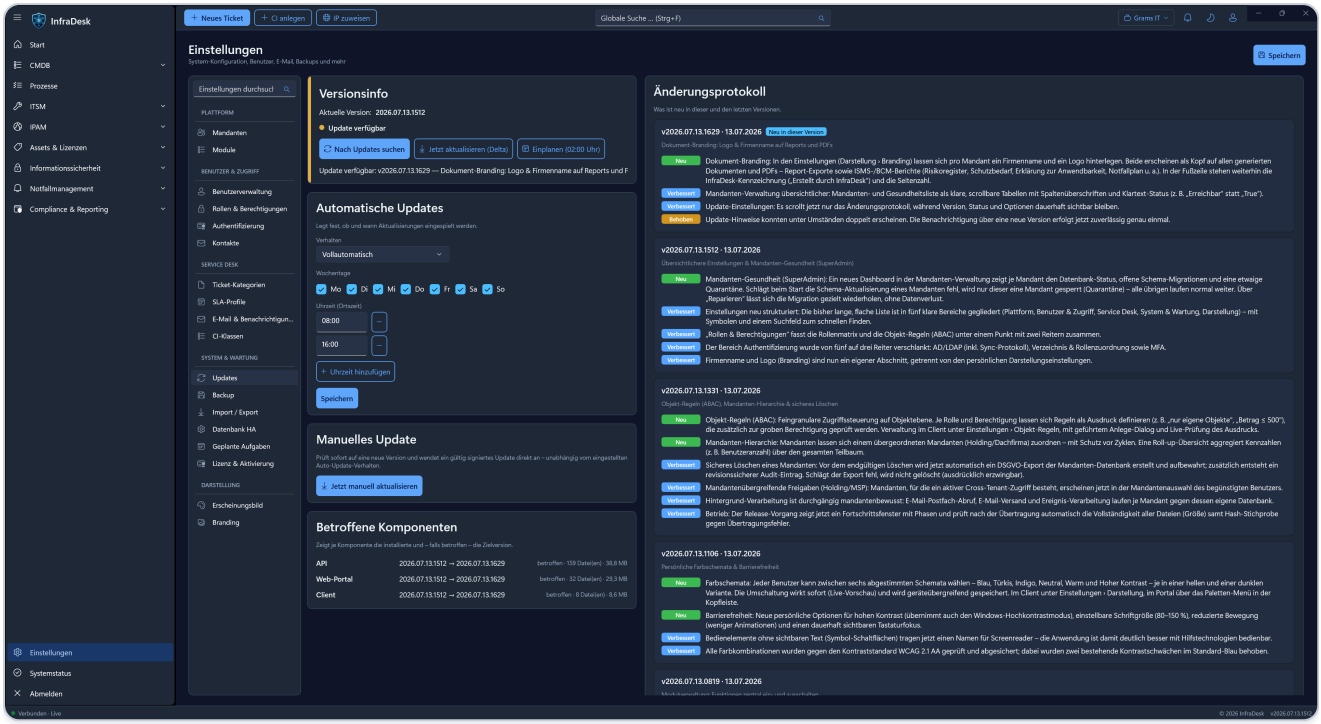

Abb. 5 – Update-Center: eine neuere Version ist verfügbar (2026.07.13.1629). Links Versionsinfo, Auto-Update-Policy mit Wartungsfenster und betroffene Komponenten; rechts das integrierte Änderungsprotokoll.

9.5 Fortschritt und geprüftes Ergebnis

Während des Laufs zeigt ein **Fortschrittsfenster** die Phasen. Weil die eigentliche Übernahme im entkoppelten Updater läuft, endet die Anzeige im Client mit „**an den Updater übergeben**" – das ist bewusst noch **kein** „erfolgreich".

Den Verlauf eines Update-Laufs zeigt Abbildung 6.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

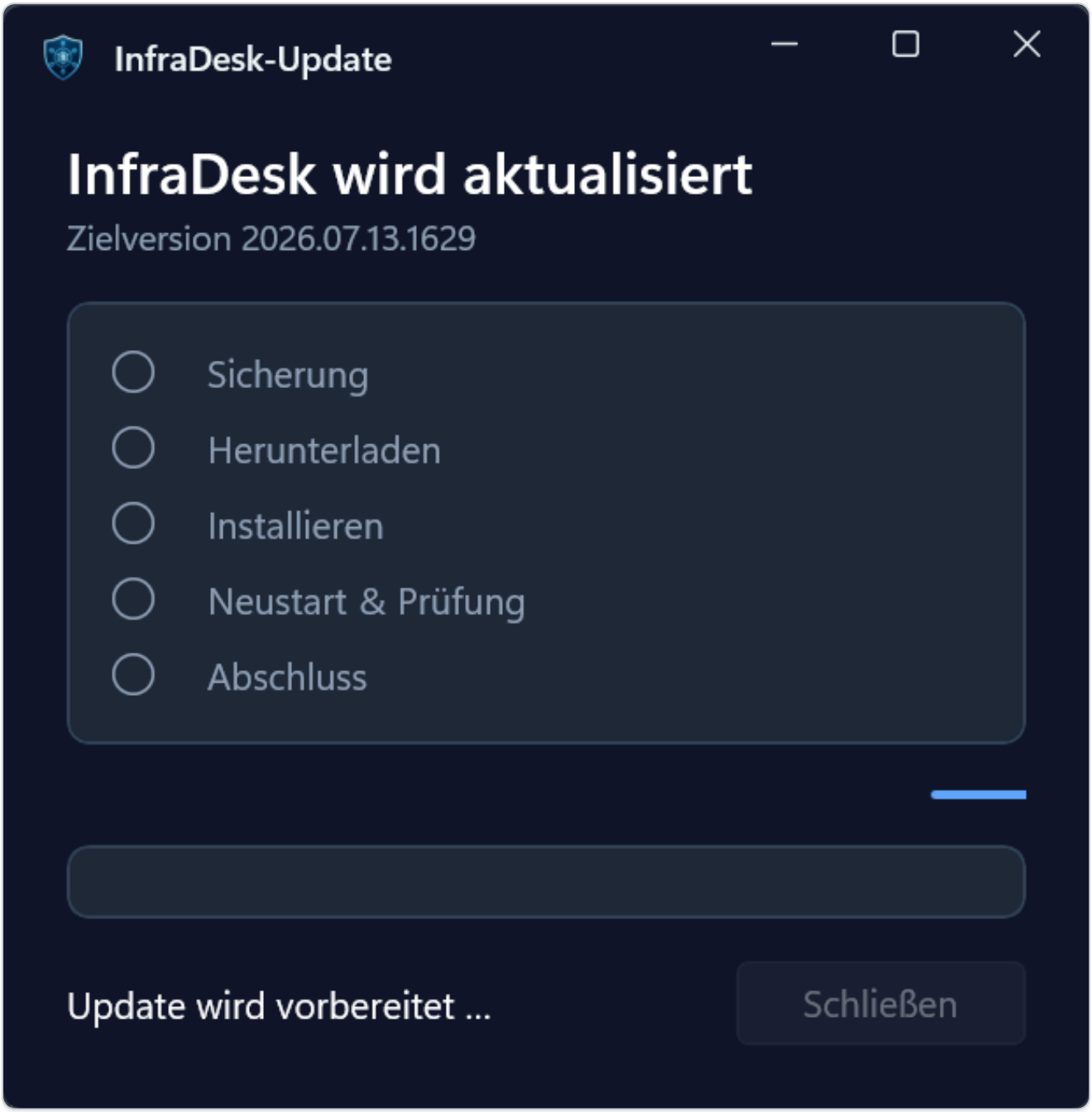

Abb. 6 – Während der Durchführung zeigt ein Fortschrittsfenster die Phasen Sicherung → Herunterladen → Installieren → Neustart & Prüfung → Abschluss (mit automatischem Rollback bei Fehlern).

Das **geprüfte Endergebnis** je Komponente (Erfolg, Rollback oder Fehler) ermittelt der Updater eigenständig und legt es in `update-status.json` ab. Der Client liest diese Datei beim nächsten Start und zeigt das Ergebnis unter **Einstellungen ▸ Updates** an – Sie sehen also den **tatsächlich verifizierten** Ausgang, nicht nur, dass ein Lauf gestartet wurde.

9.6 Update-Protokoll weitergeben

Schlägt ein Update fehl, spielt Ordivis Platform automatisch die Vorversion zurück (Rollback) und zeigt das Protokoll im Update-Fenster. Ein **Klick auf den Protokollbereich** kopiert den gesamten Text in die **Zwischenablage** – Sie können ihn damit in ein eigenes Ticket einfügen oder aus Ihrem Mailprogramm an

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

den Hersteller-Support senden (`support@ordivis.eu`). Das funktioniert bei fehlgeschlagenen wie bei erfolgreichen Läufen.

> **Ordivis Platform verschickt nichts von selbst.** Es gibt bewusst **keine** Funktion, die Fehlerberichte, Absturzdaten oder Systemmeldungen selbsttätig an den Hersteller übermittelt – keine Telemetrie, kein stiller Absturzbericht, nichts im Hintergrund. Was Ihr Haus verlässt, entscheiden ausschließlich Sie.
>
> Zwei Wege stehen Ihnen dafür offen, und **beide** setzen Ihre Handlung voraus: Sie kopieren das Protokoll und versenden es selbst (Absatz *Klick auf den Protokollbereich* über diesem Hinweis) – oder Sie nutzen den **Fehlerbericht aus dem Client** (Abschnitt 4.1). Dieser übermittelt genau das, was Sie im Dialog Punkt für Punkt bestätigen, und er läuft über das E-Mail-Postfach **Ihrer** Organisation, nicht über eines des Herstellers.

Was im Protokoll steht: die Schritte des Update-Laufs mit Zeitstempel (Sicherung, Herunterladen, Installieren, Dienst-Neustart, Health-Check) sowie die Fehlermeldung, an der der Lauf gescheitert ist – einschließlich Dateipfaden der Installation. Prüfen Sie es vor dem Weitergeben, wenn Ihre Richtlinien das verlangen; Sie sehen im Fenster exakt den Text, den Sie kopieren.

9.7 Rollendes Support-Fenster

Damit Installationen sicher und aktuell bleiben, führt Ordivis Platform ein **rollendes Support-Fenster**. Die Website veröffentlicht in `version.json` eine **Mindestversion** (`minimumVersion`) und kennzeichnet **Sicherheitsreleases** je Eintrag als **verpflichtend** (`mandatory`). Der Client leitet daraus einen gestuften **Support-Status** ab (die Produktversion ist datumsbasiert `JJJJ.MM.TT.HHMM`, das Alter also direkt ablesbar):

- **Unterstützt** – innerhalb des Fensters, kein Hinweis.
- **Alternd** – nähert sich dem Ablauf: **frühzeitige, planbare Warnung**, ein Update einzuplanen.
- **Nicht mehr unterstützt** – unter der Mindestversion **oder** ein Pflicht-(Sicherheits-)Release steht aus: ein **nicht schließbarer Hinweis** mit Direktaktion „Jetzt aktualisieren" fordert das Update bestimmt ein.

Wichtig: Die Durchsetzung erzwingt stets nur das **Update**, sie **blockiert nie den laufenden Betrieb**. Ohne Server-Kontakt (offline / air-gapped) greift kein harter Zustand – eine getrennte Installation wird niemals ausgesperrt. Die Schwelle für „alternd" ist über `Update:SoftFloorDays` (Standard 90 Tage) einstellbar; die harte Untergrenze kommt aus `minimumVersion`.

Kundennutzen: aktuelle Sicherheitspatches, Fehlerbehebungen und Funktionen; ein bekanntes, kleines Versions-Spektrum vereinfacht Support und Migrationspfad – der zulässige Rückstand ist gedeckelt.

9.8 Aufräumen des Delta-Stores (Retention)

Der Publish-Dienst räumt den Delta-Store (`updates/`) **reference-basiert** auf: Er behält die **aktuelle Version**, die **Mindestversion** und die **neuesten Releases** (`Retention:KeepReleases`, Standard 12) und entfernt ältere Versionsordner sowie jeden Blob, den **kein behaltenes Manifest referenziert**. Es wird **nie nach Alter** gelöscht (das neueste Manifest teilt sich viele langlebige Blobs mit älteren Versionen). Kunden werden dadurch **nie ausgesperrt**, weil ein Update immer auf das **vollständige neueste Manifest** zielt und

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ein weit zurückliegender Client direkt darauf springt. Das serverseitige Spiegel-Löschen ist bewusst **opt-in** (`Retention:RemotePrune`).

10 Audit-Log

Sicherheitsrelevante Aktionen sind revisionssicher im **Audit-Log** einsehbar (ID-27). Der Zugriff erfordert das Recht `audit.read`.

Was ein Eintrag festhält. Zeitpunkt, handelnde Person (Benutzerkennung und Anmeldename), die ausgeführte Aktion, Art und Kennung des betroffenen Objekts, die IP-Adresse sowie – soweit vorhanden – die geänderten Werte. Zusätzlich wird festgehalten, **worüber** die Änderung kam (Windows-Client, Self-Service-Portal, mobile App oder Schnittstelle) und von **welchem Gerät** (eine je Installation vergebene Kennung ohne Personenbezug). Das ist kein Beiwerk: Eine Inventurbestätigung, die vom Telefon im Serverraum kommt, wiegt bei einer späteren Prüfung anders als dieselbe Buchung vom Schreibtisch, und über die Gerätekennung lassen sich alle Buchungen eines verlorenen Geräts finden. Bei Einträgen aus der Zeit vor Einführung dieser beiden Angaben bleiben sie leer.

Suchen. Filtern lässt sich nach Objektart, einzelnem Objekt, handelnder Person und Zeitraum; die Ausgabe erfolgt seitenweise (höchstens 200 Einträge je Seite).

Mandantengrenze. Die Suche ist immer auf den Mandanten der angemeldeten Person begrenzt. Ein abweichender Mandant lässt sich auch dann nicht abfragen, wenn er in der Anfrage mitgegeben wird.

Ausgeben. Über **Exportieren** wird das Audit-Log mit den *aktuell eingestellten Filtern* als CSV-Datei ausgegeben (Semikolon getrennt, UTF-8 – Excel öffnet sie direkt). Dafür ist das eigene Recht `audit.export` nötig; `audit.read` allein genügt nicht, denn eine ausgegebene Datei verlässt das System und lässt sich nicht zurückholen. Ohne das Recht ist die Schaltfläche nicht sichtbar, und der Abruf wird auch dann abgewiesen, wenn er direkt an die Schnittstelle gerichtet wird.

Die Datei beginnt mit drei Kommentarzeilen: Erzeugungszeitpunkt, abgedeckter Zeitraum und die Angabe, ob der Auszug **vollständig** ist. Aus Rücksicht auf den Arbeitsspeicher des Servers ist die Ausgabe auf 50.000 Einträge begrenzt; wird diese Grenze erreicht, steht das ausdrücklich im Kopf der Datei – grenzen Sie in diesem Fall den Zeitraum ein und geben Sie mehrere Teilauszüge aus.

11 Modulverwaltung (Funktionsmodule ein-/ausschalten)

> 💻 **Client:** *Fußbereich ▸ Einstellungen ▸ Module* · 🌐 **Portal:** *Verwaltung ▸ Mandanten und Module* · 🔑 `tenant.manage`

Nicht jede Organisation braucht alle Funktionen. Der **SuperAdmin** kann einzelne Funktionsmodule zentral aktivieren oder deaktivieren.

🔒 **Pflichtreihenfolge** — der globale Standard wirkt nur dort, wo **kein** Mandanten-Override gesetzt ist. Entfernen Sie deshalb zuerst mit *Global übernehmen* den Override, bevor Sie sich über einen wirkungslosen globalen Schalter wundern.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Öffnen Sie die Modulverwaltung.
2. Setzen Sie je Modul den **globalen Standard**.
3. Wechseln Sie bei Bedarf auf einen einzelnen **Mandanten** und setzen Sie dort einen abweichenden Wert.
4. Um einen Override wieder zu entfernen, klicken Sie auf *Global übernehmen*.

✅ **Ergebnis:** Ein deaktiviertes Modul verschwindet aus der Navigation von Client **und** Portal; seine REST-Endpunkte antworten mit **HTTP 404**.

> **Achtung:** Ein abgeschaltetes Modul löscht keine Daten – es blendet sie aus. Beim Wiedereinschalten ist alles unverändert da. Wer Daten wirklich loswerden will, ist hier falsch.

Geltungsbereich. Je Modul lässt sich ein **globaler Standard** setzen; zusätzlich kann pro **Mandant** ein abweichender Wert (Override) hinterlegt werden. Es gilt: *Mandanten-Override sticht globalen Standard, sonst globaler Standard, sonst „aktiv“*. Mit „Global übernehmen“ wird ein Mandanten-Override wieder entfernt.

Wirkung. Ein deaktiviertes Modul verschwindet aus der Navigation von Client **und** Portal; die zugehörigen REST-Endpunkte antworten mit **HTTP 404** (das Modul „existiert“ nach außen nicht). Die Absicherung wirkt serverseitig – nicht nur durch Ausblenden in der Oberfläche.

Pflichtmodule (immer aktiv, nicht abschaltbar): CMDB, Identität & Zugriff, Audit-Log, Benachrichtigungen, Produktaktivierung, Lizenzierung sowie Berichte & Dashboards.

Abschaltbare Module: Service Desk, Wissensdatenbank (eigenständig), IPAM, Discovery, Verträge & Assets, Notfallmanagement (BCM), Wiederanlaufplanung (DR), ISMS (Security), Workflow und Datenschutz (VVT).

> Berechtigung: Die Modulverwaltung ist der Plattform-Ebene vorbehalten und erfordert das Recht `tenant.manage`.

12 Plattform-Überwachung

Das Dashboard **Plattform-Überwachung** zeigt den Zustand der Installation auf einen Blick. Es aktualisiert sich automatisch alle 10 Sekunden, solange die Seite geöffnet ist.

Bereich	Kennzahlen
PostgreSQL	Version und Rolle, aktive Verbindungen gegen Maximum (inkl. Auslastung), Cache-Trefferquote, Commits/Rollbacks, lang laufende Abfragen, Gesamtgröße
Sitzungen	aktive Sitzungen, aktive Benutzer, Anmeldungen der letzten 24 h
Je Mandant	Status, Datenbankgröße, aktive Sitzungen

Worauf zu achten ist: Eine dauerhaft hohe **Verbindungsauslastung** oder eine fallende **Cache-Trefferquote** deuten auf Engpässe hin; **lang laufende Abfragen** sind der übliche erste Hinweis bei Trägheit. Ist ein

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Mandant **nicht erreichbar**, wird das ausgewiesen statt still übergangen – die übrigen Kennzahlen bleiben gültig.

„–" heißt „nicht ermittelbar", nicht „null". Kann eine Größe nicht erhoben werden, zeigt das Dashboard einen Strich und liefert die übrigen Kennzahlen weiter – statt die ganze Seite leer zu lassen. Die **Datenbankgrößen** sind dabei ein Sonderfall: Sie erhebt PostgreSQL, indem es je Datenbank deren Verzeichnisbaum durchgeht, was mit der Zahl der Mandanten teurer wird. Sie werden deshalb nur alle paar Minuten neu bestimmt und dazwischen aus einem Zwischenspeicher bedient – die Zahl kann also etwas älter sein als der Rest der Anzeige. Gezählt wird ausschließlich, was zu Ordivis Platform gehört (Haupt- und Mandanten-Datenbanken); fremde Datenbanken auf demselben Server bleiben außen vor.

Berechtigung: `platform.monitor` (Plattform-Ebene).

13 Sicherheitsereignisse im Windows-Ereignisprotokoll

Zusätzlich zum Audit-Log (Abschnitt 10) schreibt Ordivis Platform sicherheitsrelevante Vorgänge in das **Windows-Ereignisprotokoll** unter der eigenen Quelle **„Ordivis Platform"** im Protokoll *Anwendung*. Damit lassen sie sich ohne Zusatzaufwand an ein vorhandenes **SIEM** anbinden, das Windows-Logs bereits einsammelt.

Erfasst werden unter anderem: Anmeldungen und **fehlgeschlagene** Anmeldungen, Kennwortänderungen, Änderungen an Rollen und Rechten, Dienst-, Update- und Lizenzereignisse.

Welche Ereignisse geschrieben werden, konfigurieren Sie auf Plattform-Ebene – so bleibt das Protokoll auf das beschränkt, was Ihr SIEM auswerten soll.

Das Audit-Log wird dadurch nicht ersetzt. Es bleibt die vollständige, revisionssichere Historie in der Anwendung; das Ereignisprotokoll ist die Schnittstelle nach außen.

Berechtigung: `platform.eventlog.manage` (Plattform-Ebene).

13.1 Ereignis-Referenz für Ihr SIEM

Jeder Eintrag trägt eine feste **Ereigniskennung**. Sie ist der Ankerpunkt für Regeln und Auswertungen und ändert sich nicht: Neue Ereignisse bekommen die nächste freie Nummer ihrer Kategorie, bestehende werden nie umnummeriert. Die Nummer folgt dem Schema *Kategorie* × *1000* + *laufende Nummer*; die Kategorie erscheint im Ereignis zusätzlich als Windows-*Kategorie*.

Authentifizierung

ID	Ereignis	Schweregrad	Bedeutung
1001	`LoginSucceeded`	Information	Anmeldung erfolgreich
1002	`LoginFailed`	Warnung	Anmeldung fehlgeschlagen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ID	Ereignis	Schweregrad	Bedeutung
1003	`Logout`	Information	Abmeldung
1004	`AccountLockedOut`	Warnung	Konto wegen zu vieler Fehlversuche gesperrt
1005	`MobileDeviceRevoked`	Warnung	Mobilgerät abgemeldet (Widerruf durch den Benutzer)

Konto

ID	Ereignis	Schweregrad	Bedeutung
2001	`PasswordChanged`	Information	Kennwort geändert
2002	`PasswordReset`	Warnung	Kennwort administrativ zurückgesetzt
2003	`UserCreated`	Information	Benutzerkonto angelegt
2004	`UserDeactivated`	Warnung	Benutzerkonto deaktiviert

Rollen & Rechte (RBAC)

ID	Ereignis	Schweregrad	Bedeutung
3001	`RoleAssignmentChanged`	Warnung	Rollenzuweisung geändert
3002	`PermissionChanged`	Warnung	Berechtigung geändert
3003	`SuperAdminAction`	Warnung	Plattform-(SuperAdmin-)Aktion

Dienst-Lebenszyklus

ID	Ereignis	Schweregrad	Bedeutung
4001	`ServiceStarted`	Information	Dienst gestartet
4002	`ServiceStopped`	Information	Dienst gestoppt
4003	`HealthCritical`	Fehler	Health-Prüfung kritisch

Update

ID	Ereignis	Schweregrad	Bedeutung
5001	`UpdateApplied`	Information	Update angewendet
5002	`UpdateFailed`	Fehler	Update fehlgeschlagen
5003	`UpdateRolledBack`	Warnung	Update zurückgerollt

Lizenz

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ID	Ereignis	Schweregrad	Bedeutung
6001	`LicenseActivated`	Information	Lizenz aktiviert
6002	`LicenseExpired`	Warnung	Lizenz abgelaufen
6003	`LicenseTampering`	Fehler	Lizenz-Manipulationsverdacht

Backup

ID	Ereignis	Schweregrad	Bedeutung
7001	`BackupSucceeded`	Information	Backup erfolgreich
7002	`BackupFailed`	Fehler	Backup fehlgeschlagen

System

ID	Ereignis	Schweregrad	Bedeutung
8001	`CriticalError`	Fehler	Kritischer Systemfehler
8002	`MigrationRun`	Information	Datenbank-Migration ausgeführt
8003	`EventLogConfigChanged`	Warnung	Sicherheitsprotokoll-Konfiguration geändert
8004	`PlatformThresholdBreached`	Warnung	Plattform-Schwellwert überschritten

Zu jedem Eintrag gehören strukturierte **Felder** im Ereignistext, u. a. wer die Aktion ausgelöst hat (`AusgeloestVon`, `AusloeserId`), von wo (`QuellIP`), der betroffene Mandant (`MandantId`) und – je nach Ereignis – der betroffene Benutzer (`BetroffenerBenutzer`) oder die geänderte Rolle. Bei `PermissionChanged` stehen zusätzlich `Vorher` und `Nachher` mit den Rechte-Schlüsseln, sodass im Nachhinein nachvollziehbar ist, **was** sich geändert hat – nicht nur, **dass** sich etwas geändert hat.

13.2 Kanal an ein SIEM weiterleiten

Die Ereignisse liegen im Protokoll **Anwendung** unter der Quelle **Ordivis Platform** – also dort, wo Ihre vorhandene Sammlung sie ohnehin abholt. Zwei übliche Wege:

- **Windows Event Forwarding (WEF)** ohne Zusatzsoftware: Auf dem Collector ein Abonnement anlegen und in der Ereignisauswahl (XPath) nach der Quelle filtern:

```
xml <QueryList> <Query Id="0" Path="Application"> <Select Path="Application">*[System[Provider[@Name='Ordivis Platform']]]</Select> </Query> </QueryList>
```

- **SIEM-Agent** (Splunk UF, Elastic Winlogbeat, Wazuh o. ä.): Kanal `Application` einsammeln und auf den Provider `Ordivis Platform` filtern.

> **Umfang zuerst festlegen.** Was überhaupt geschrieben wird, steuern Sie in der Anwendung (Plattform-Ebene): global, je Kategorie und über einen Mindest-Schweregrad. Es ist sinnvoller, dort zu begrenzen, als später im SIEM zu verwerfen – was nicht geschrieben wird, kostet auch keine Lizenz.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

13.3 Warum diese Tabelle nicht von Hand gepflegt wird

Die Referenz oben wird aus dem Ereignis-Katalog der Anwendung erzeugt und durch einen Test gegen ihn abgeglichen. Eine abgetippte Tabelle driftet: Ein neues Ereignis wird ergänzt, die Doku nicht – und Ihr SIEM-Regelwerk beruht ab da auf einer Referenz, die nicht mehr stimmt. Weicht die Tabelle vom Katalog ab, schlägt der Test fehl.

14 Identitätsanbieter (SSO/OIDC)

Unter **Einstellungen → Benutzer & Zugriff → Identitätsanbieter (SSO)** werden OIDC-Anbieter je Mandant gepflegt. Sichtbar mit `system.security.manage` – demselben Recht, das die schreibenden Aufrufe verlangen.

Einzutragen sind Autorität (Issuer), Client-Kennung, Rückleitungsadresse und die Geltungsbereiche. Die **E-Mail-Domänen** entscheiden, ob die Anmeldeseite den Anbieter von sich aus vorschlägt: Ohne Eintrag muss er von Hand gewählt werden.

Probe vor dem Scharfschalten. Die Schaltfläche „Probe ausführen" ruft die Angaben des Anbieters ab und zeigt Issuer, Endpunkte, die Zahl der Signaturschlüssel und ob PKCE mit S256 angeboten wird. Fehlt eine Abmeldeadresse oder S256, meldet sie **Warnung** statt Erfolg – beides hat Folgen im Betrieb: Ohne Abmeldeadresse endet ein Abmelden nur in Ordivis, die Sitzung beim Anbieter bleibt bestehen; ohne S256 startet gar keine Anmeldung.

> **Achtung – Aussperren möglich.** Der Schalter „Anmeldung nur über diesen Anbieter" deaktiviert die Kennwortanmeldung. Ist der Anbieter nicht erreichbar, kommt niemand mehr herein. Vor dem Einschalten die Probe erfolgreich durchführen **und** ein Notfallkonto bereithalten, das den Anbieter nicht braucht.

Das **Client-Geheimnis** wird nie angezeigt – die Seite meldet nur, ob eines hinterlegt ist. Ein leeres Feld bedeutet „unverändert" und löscht nichts.

Gruppen auf Rollen abbilden. Der Anspruch mit den Gruppen (meist `groups` oder `roles`) wird eingetragen, darunter die Zuordnungen. Die **Vorschau** rechnet an einem Beispielkonto vor, welche Rollen entstünden – ohne Speichern. Zeigt eine Zuordnung auf eine gelöschte Rolle, gilt die Gruppe als nicht zugeordnet; der Server weist solche Zuordnungen beim Speichern ab.

15 Zugriffsüberprüfung (Access Review)

Unter **Einstellungen → Benutzer & Zugriff → Zugriffsüberprüfung**; Recht `identity.permissions.manage`. Die wiederkehrende Bestätigung, wer welche Rolle hat, ist die Kontrolle, die ISO 27001 unter **A.5.18** verlangt.

Ein Durchlauf entsteht als Entwurf mit Zeitraum. Beim **Starten** sammelt er alle Rollenzuweisungen des Mandanten ein und friert sie ein – ein zweiter Start fügt nichts hinzu, damit nachvollziehbar bleibt, worüber entschieden wurde. Jede Position wird einzeln bestätigt oder entzogen, mit Begründung; Person und Rolle stehen mit Klarnamen da.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Abgeschlossen werden kann erst, wenn keine Position mehr offen ist. Ein abgeschlossener Durchlauf mit offenen Positionen wäre ein Nachweis, der nichts belegt – der Server weist das ab, nicht nur die Oberfläche.

Prüfer und Ersteller stammen ausschließlich aus der Anmeldung und lassen sich nicht setzen. Bei einer Zugriffsüberprüfung ist der Nachweis, *wer* bestätigt hat, der Kern der Kontrolle.

16 Mandantenübergreifende Freigaben

Unter **Einstellungen** → **Plattform** → **Mandanten-Freigaben**; Recht `tenant.grant.manage` (Plattform-Super-Admin). Für Holding-Strukturen und Dienstleistungsverträge: Eine Person erhält zeitlich begrenzten Zugriff auf einen fremden Mandanten. **Die Vorgabe ist aus** – ohne Freigabe bleibt jeder auf seine eigenen Mandanten beschränkt.

Die Rechte werden aus dem Katalog gewählt, nicht getippt: Ein Recht, das es nicht gibt, fiele erst beim Zugriff auf, und dann still. Sie müssen im gewährenden Mandanten existieren.

> **Achtung – ohne Enddatum läuft eine Freigabe nie ab.** Die Liste weist solche Einträge ausdrücklich aus. Für einen Dienstleistungsvertrag gehört das Vertragsende in das Feld „Gültig bis“.

Widerrufene und abgelaufene Freigaben bleiben in der Liste stehen; ein Widerrufsknopf erscheint nur, solange die Freigabe tatsächlich wirkt. Jeder Zugriff über eine Freigabe steht im Audit-Log.

17 Automatisierungsregeln für Vorgänge

Unter **Vorgänge** → **Werkzeuge** → **Automatisierungsregeln**; Recht `ticketing.config.manage` zum Pflegen. Regeln der Form Bedingung → Aktion greifen beim Anlegen eines Vorgangs.

Als Bedingung dienen Vorgangsart, Kategorie, ein Textstück im Betreff und die Domäne des Melders; als Aktion das Setzen von Priorität, Support-Gruppe, Kategorie oder eines Mitlesers. Die Regeln laufen in der eingetragenen Reihenfolge; die erste zutreffende mit **„Kette anhalten“** beendet die Verarbeitung.

> **Nicht mit den Routing-Regeln verwechseln.** Beide Regelwerke laufen beim Anlegen, sind aber getrennt: Die Routing-Regeln bilden ausschließlich Kategorie → Support-Gruppe ab.

Die Liste zeigt Bedingung und Aktion ausgeschrieben. Eine Regel ohne Aktion – und ebenso eine, die nur die Kette anhält – wird als wirkungslos ausgewiesen: Die zweite ändert nichts, verdeckt aber alle nachfolgenden Regeln.

18 Protokollierung, Diagnose und Support-Paket

Unter **Einstellungen** → **System & Wartung** → **Protokolle**; nur Plattform-Ebene. Hier werden Speicherort, Aufbewahrung und Rotation gesetzt sowie die Mindeststufe je Bereich – getrennt nach Technik, Anwendungen und Fachmodulen. **Die Umstellung wirkt ohne Dienst-Neustart.**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Das **Support-Paket** steht bewusst ganz oben auf derselben Seite: Es sammelt Protokolle aller Anwendungen, Update-Protokolle, den Steckbrief der Anlage, eine geschwärzte Konfiguration und einen Auszug des Windows-Ereignisprotokolls in ein AES-256-verschlüsseltes Archiv.

> **Das Kennwort gehört nicht in dieselbe Nachricht wie das Archiv.**

Was aufgenommen und was ausgelassen wurde, steht in der Datei `01-INHALT.txt` im Archiv.

19 Datenschutz: Betroffenenrechte und AVV-Register

Zwei Seiten unter **Datenschutz** in der Hauptnavigation.

Betroffenenrechte (Recht `privacy.manage`) ist das Werkzeug des Datenschutzbeauftragten, keine Selbstbedienung: Über die E-Mail-Adresse werden Konten, Kontakte und Vorgänge einer Person zusammengestellt. Die Ausgabe erfolgt als **JSON** – Art. 20 DSGVO verlangt ein maschinenlesbares Format. Steht zu einer Adresse nichts, ist auch das eine gültige Auskunft nach Art. 15 und der betroffenen Person mitzuteilen.

Die **Löschung nach Art. 17 anonymisiert**, sie löscht nicht: Vorgänge und Nachweise bleiben auswertbar, die Person ist darin nicht mehr erkennbar. Sie ist erst möglich, nachdem die Auskunft erstellt wurde – der Schritt lässt sich nicht rückgängig machen, und wer vorher gesehen hat, was daran hängt, entscheidet anders.

AVV-Register (Rechte wie am Verarbeitungsverzeichnis: `datenschutz.vvt.read` zum Lesen, `.edit` zum Pflegen) führt die Auftragsverarbeitungsverträge nach Art. 28. Abgelaufene stehen oben und farblich abgesetzt – ein abgelaufener Vertrag zu einem aktiven Auftragsverarbeiter ist ein Befund und zählt in der Kachel „AVV abgelaufen“ des Verarbeitungsverzeichnisses mit. Der Dienstleister wird aus der CMDB benannt und dort gepflegt, nicht hier.

20 Eigenes Postfach für den Exchange-Kalender

Keine Administrationsaufgabe, aber hier erwähnt, weil danach gefragt wird: Jeder Benutzer gibt sein Postfach **selbst** frei – im Portal unter **Mein Konto**, im Arbeitsplatz-Client unter **Einstellungen → Mein Postfach (Exchange)**.

Ohne diese Freigabe wird kein Postfach gelesen, auch nicht das eines Administrators. Das Dienstkonto mit `ApplicationImpersonation` könnte technisch jedes Postfach der Organisation öffnen – das ist Exchanges Modell und dort nicht abwählbar; begrenzt wird es im Produkt. Einschalten und Widerruf stehen im Sicherheitsprotokoll, und widerrufene Freigaben bleiben in der Liste: Sie sind der Nachweis darüber, in welchem Zeitraum zugegriffen werden durfte.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

21 Wichtige Rechte für Administration

Recht	Erlaubt
`system.config.manage`	Systemeinstellungen verwalten (= „Administrator“)
`system.security.manage`	Sicherheits-/Anmeldedaten verwalten
`identity.users.manage` / `identity.roles.manage`	Benutzer bzw. Rollen verwalten (ID-41)
`tenant.manage`	Mandanten- **und** Modulverwaltung (Plattform-/SuperAdmin)
`platform.monitor`	Plattform-Überwachung ansehen
`platform.eventlog.manage`	Windows-Ereignisprotokoll konfigurieren
`identity.permissions.manage`	Zugriffsüberprüfung durchführen (Kap. 15)
`tenant.grant.manage`	Mandantenübergreifende Freigaben erteilen (Kap. 16)
`ticketing.config.manage`	Automatisierungs- und Routing-Regeln pflegen (Kap. 17)
`privacy.manage`	Betroffenenrechte bearbeiten (Kap. 19)

Verwandte Themen

- **ID-41 – Modul Identität, Benutzer & Rollen** – Benutzer, Rollen und Anmeldeverfahren
- **ID-42 – Modul Mandantenfähigkeit** – Mandanten anlegen und trennen
- **ID-43 – Modul Produktlizenzierung** – Lizenz aktivieren und Limits prüfen
- **ID-31 – Modul Benachrichtigungen & E-Mail** – E-Mail-Versand und Posteingang
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – alles, was am Server selbst passiert

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · IDENTITÄT & RBAC

Modul Identität, Benutzer & Rollen (RBAC)

Benutzer anlegen, Rollen zuweisen, MFA, Passwörter, Rechtematrix

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-41 Identität

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Identität, Benutzer & Rollen (RBAC)

Nicht jeder in einer Organisation soll alles dürfen. Die Auszubildende im ersten Monat braucht andere Möglichkeiten als die Administratorin, und die Buchhaltung hat in der IT-Dokumentation wenig verloren. Würde man jeder Person ihre Rechte einzeln zuteilen, wäre das bei mehr als einer Handvoll Leuten kaum noch zu überblicken und fehleranfällig obendrein.

Deshalb arbeitet Ordivis Platform *rollenbasiert*, im Fachjargon RBAC für „Role-Based Access Control“. Der Gedanke ist einfach: Berechtigungen (Rechte) werden nicht direkt an Personen vergeben, sondern an Rollen wie „Agent“, „Administrator“ oder „Nur-Lesen“. Eine Person bekommt eine oder mehrere Rollen und erbt damit genau deren Rechte. Ändert sich der Zuschnitt einer Rolle, wirkt das sofort für alle, die sie haben. Dieses Modul verwaltet die drei Bausteine dahinter: die Konten (Benutzer), die Rollen und die Rechte.

INHALT

1 Grundprinzip: permission-basiert
2 Lokalen Benutzer anlegen
3 Stammdaten & Status pflegen
4 Benutzername & Kennwort
4.1 Sperre nach Fehlversuchen
4.2 Nur eine Anmeldung gleichzeitig
5 Rollen & Rechtematrix
6 Mehr-Faktor-Authentifizierung (MFA)
6.1 Einrichten (Selbstbedienung)
6.2 Anmelden mit dem zweiten Faktor
6.3 Abschalten verlangt eine Bestätigung
7 Administrator-Limit (Lizenz)
8 Verzeichnisbenutzer (AD/LDAP)
8.1 Welche Verzeichnisattribute übernommen werden
8.2 Wie aus AD-Gruppen Rollen werden
Verwandte Themen

1 Grundprinzip: permission-basiert

Das Anmelde-Token (JWT) enthält ausschließlich `permission`**-Claims** – keine Rollennamen. Die Sichtbarkeit von Menüpunkten und Aktionen richtet sich nach diesen Rechten. Serverseitig sichert `[RequiresPermission]` jeden Endpunkt ab.

2 Lokalen Benutzer anlegen

💻 **Client:** *Fußbereich ▸ Einstellungen ▸ Benutzerverwaltung ▸ Neuer Benutzer* · 🌐 **Portal:** *Verwaltung ▸ Benutzer und Rollen* · 🔑 `identity.users.manage`

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

🔒 **Pflichtreihenfolge** — der Benutzer wird **passwortlos** angelegt und kann sich damit nicht anmelden. Schritt 4 gehört zwingend dazu; ohne ihn ist der Vorgang nicht abgeschlossen, sondern halb erledigt.

1. Öffnen Sie die Benutzerverwaltung und klicken Sie auf *Neuer Benutzer*.
2. Erfassen Sie *Login*, *Anzeigename*, *E-Mail* und den *Mandanten*.
3. Weisen Sie die *Rollen* zu – sie entscheiden, was die Person sieht und darf (ID-44).
4. Speichern Sie und vergeben Sie anschließend über *Passwort zurücksetzen* ein **erstes Kennwort**.
5. Geben Sie das Kennwort auf einem anderen Weg als per E-Mail an die Person weiter.

✅ **Ergebnis:** Die Person kann sich anmelden und ihr Kennwort danach selbst ändern (ID-10, Abschnitt 3).

> **Wichtig:** Ein neu angelegter Benutzer kann sich erst anmelden, nachdem ein Administrator ein **Kennwort zurückgesetzt** (vergeben) hat. Das ist der häufigste Grund für die Meldung, die Anmeldung funktioniere nicht.

3 Stammdaten & Status pflegen

- **Ändern:** Anzeigename, **Vorname**, **Nachname**, **Funktion/Titel**, E-Mail, Abteilung, Telefon. Vor- und Nachname speisen u. a. die E-Mail-Signatur des Service Desks (Nachname dort aus Datenschutzgründen gekürzt).
- **Aktivieren/Deaktivieren:** sperrt bzw. reaktiviert den Zugang.
- **Passwort zurücksetzen:** vergibt ein neues Kennwort.
- **MFA zurücksetzen:** entfernt den zweiten Faktor (Neu-Einrichtung nötig).
- **Sessions:** aktive Anmeldungen einsehen/beenden.

4 Benutzername & Kennwort

Der **Benutzername** ist **case-insensitiv** (Anmeldung unabhängig von Groß-/Kleinschreibung), das **Kennwort** ist case-sensitiv. Abbildung 1 zeigt die Anmeldung im Ganzen – einschließlich Sperre (Abschnitt 4.1) und zweitem Faktor (Abschnitt 6).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

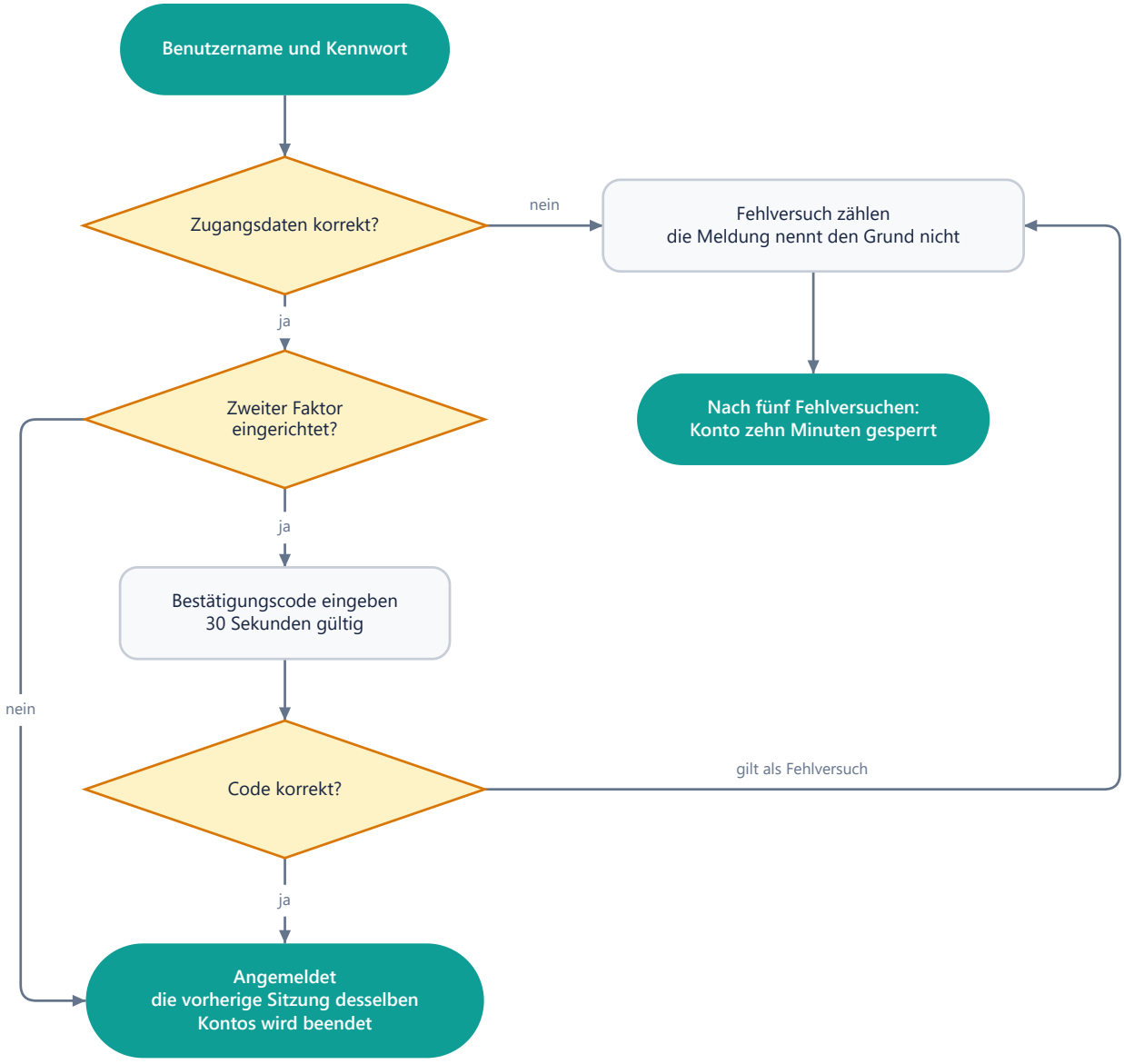

Abb. 1 – Anmeldung mit Kennwort und zweitem Faktor. Ein falscher Bestätigungscode läuft in dieselbe Sperre wie ein falsches Kennwort.

4.1 Sperre nach Fehlversuchen

Nach **fünf** aufeinanderfolgenden Fehlversuchen wird das Konto für **zehn Minuten** gesperrt; danach ist eine Anmeldung wieder möglich, ohne dass jemand eingreifen muss. Ein erfolgreicher Versuch setzt den Zähler zurück. Beide Werte sind in der Serverkonfiguration einstellbar (`Login:BruteForceMaxAttempts`, `Login:BruteForceLockMinutes`).

Eine Administratorin oder ein Administrator muss eine solche Sperre nicht aufheben. Wer nicht warten möchte, kann über **Passwort zurücksetzen** ein neues Kennwort vergeben.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> Die Anmeldemaske nennt bei falschen Zugangsdaten bewusst **nicht**, ob es das Konto überhaupt gibt – und antwortet in beiden Fällen gleich schnell. Andernfalls ließe sich von außen herausfinden, welche Benutzernamen existieren. Jeder Fehlversuch und jede ausgelöste Sperre stehen im Windows-Sicherheits- protokoll (siehe Administrationshandbuch) und lassen sich damit an ein SIEM anbinden.

4.2 Nur eine Anmeldung gleichzeitig

Eine neue Anmeldung **beendet die vorherige Sitzung desselben Kontos**. Wer sich am zweiten Gerät anmeldet, wird am ersten beim nächsten Zugriff abgemeldet. Das ist beabsichtigt: Ein Konto, das an mehreren Orten gleichzeitig offen ist, lässt sich im Nachhinein keiner Person mehr zuordnen.

Die Sitzung selbst läuft nach **zwölf Stunden ohne Zugriff** ab; das kurzlebige Zugriffstoken wird währenddessen alle **15 Minuten** im Hintergrund erneuert (`Identity:Jwt:RefreshTokenExpiryHours`, `Identity:Jwt:AccessTokenExpiryMinutes`). Die zwölf Stunden sind eine Frist der Ruhe, keine Höchstdauer: Jede Erneuerung setzt sie neu, ein geöffneter Client hält die Sitzung also offen. Sie läuft ab, wenn zwölf Stunden lang kein Zugriff erfolgt — Anwendung geschlossen, Rechner aus oder im Ruhezustand. Aktive Sitzungen lassen sich in der Benutzerverwaltung einsehen und gezielt beenden (Abschnitt 3).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Auf dem Server einstellen. Beide Werte stehen als Maschinen-Umgebungsvariable (`Identity__Jwt__RefreshTokenExpiryHours` , `Identity__Jwt__AccessTokenExpiryMinutes`) und werden vom Installer gesetzt. Sie haben Vorrang vor `appsettings.Production.json` — und sind zugleich der einzige Weg, der eine bestehende Installation erreicht, denn Updates lassen `appsettings*.json` bewusst unangetastet. Nach einer Änderung den Dienst `InfraDesk.API` neu starten.

Das Web-Portal hat eine eigene, deutlich kürzere Frist. Es bringt eine zweite Uhr mit — die Laufzeit seines Anmelde-Cookies, einstellbar über `Session:LifetimeMinutes` (Vorgabe **30 Minuten**). Sie gilt unabhängig von den zwölf Stunden oben: Wer im Portal arbeitet, wird nach einer halben Stunde ohne Zugriff zur Anmeldung geschickt. Das ist Absicht — das Portal steht Endanwendern offen und läuft auch an gemeinsam genutzten Rechnern. Siehe Betriebshandbuch, Abschnitt 3.5.

Warum wurde ich abgemeldet? Das Portal unterscheidet die beiden Fälle seit `2026.08.07.2104` und benennt sie auch so: *„Diese Sitzung wurde beendet"* steht für eine Anmeldung an anderer Stelle (oder eine gezielt beendete Sitzung), *„Ihre Anmeldung ist abgelaufen"* für das Ende der zwölf Stunden. Vorher meldete die Schnittstelle beides mit demselben Text, und das Portal sprach immer von „abgelaufen" – was beim häufigeren Fall, der Zweitanmeldung, in die Irre führte. Für die Fehlersuche steht der Grund zusätzlich im Protokoll des Portaldienstes.

Der Arbeitsplatz-Client folgt seit `2026.08.08.1302` **.** Zuvor räumte er die beendete Sitzung stillschweigend weg: Das Fenster blieb offen, und jede Seite scheiterte einzeln mit ihrer eigenen Meldung („… konnte nicht geladen werden: Unauthorized"). Jetzt kehrt er zur Anmeldemaske zurück und nennt denselben Grund wie das Portal.

⚠️ Zu beachten: Das **Zugriffstoken** bleibt nach dem Widerruf noch bis zu 15 Minuten gültig. Die Abmeldung am ersten Gerät wird deshalb nicht sofort sichtbar, sondern erst beim nächsten Erneuern.

Eine Störung meldet nicht ab. Antwortet die Schnittstelle beim Erneuern mit einer Überlastung (429), einem Serverfehler oder gar nicht, bleibt die Anmeldung bestehen und der nächste Versuch erneuert erneut. Nur ein **zurückgewiesenes** Erneuerungs-Token beendet die Sitzung. Andernfalls würde eine kurze Lastspitze die halbe Belegschaft aus der Arbeit werfen.

5 Rollen & Rechtematrix

💻 **Client:** *Fußbereich ▸ Einstellungen ▸ Benutzerverwaltung* · 🌐 **Portal:** *Verwaltung ▸ Benutzer und Rollen* · 🔑 `identity.roles.manage`

Auf der Benutzer-Detailseite werden **Rollen** über Spalten mit Haken zugewiesen. Eine **Rollen-Matrix** zeigt, welche Rolle welche Rechte besitzt.

So weisen Sie einem Benutzer eine Rolle zu:

1. Öffnen Sie die Benutzerverwaltung und suchen Sie den Benutzer.
2. Setzen Sie in seiner Zeile den Haken in der Spalte der gewünschten Rolle.
3. Speichern Sie.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

✅ **Ergebnis:** Die Rolle greift, sobald sich der Benutzer **neu anmeldet** – die Rechte stehen im Token, und das wird bei der Anmeldung ausgestellt (ID-44, Abschnitt 5.2).

> **Wichtig:** Vergeben Sie Rechte nach dem Prinzip der **minimalen Berechtigung**. Mehrere Rollen addieren sich; die effektiven Rechte sind ihre Vereinigung, nie ihr Durchschnitt.

Abbildung 2 zeigt die Zuweisung als angehakte Spalten.

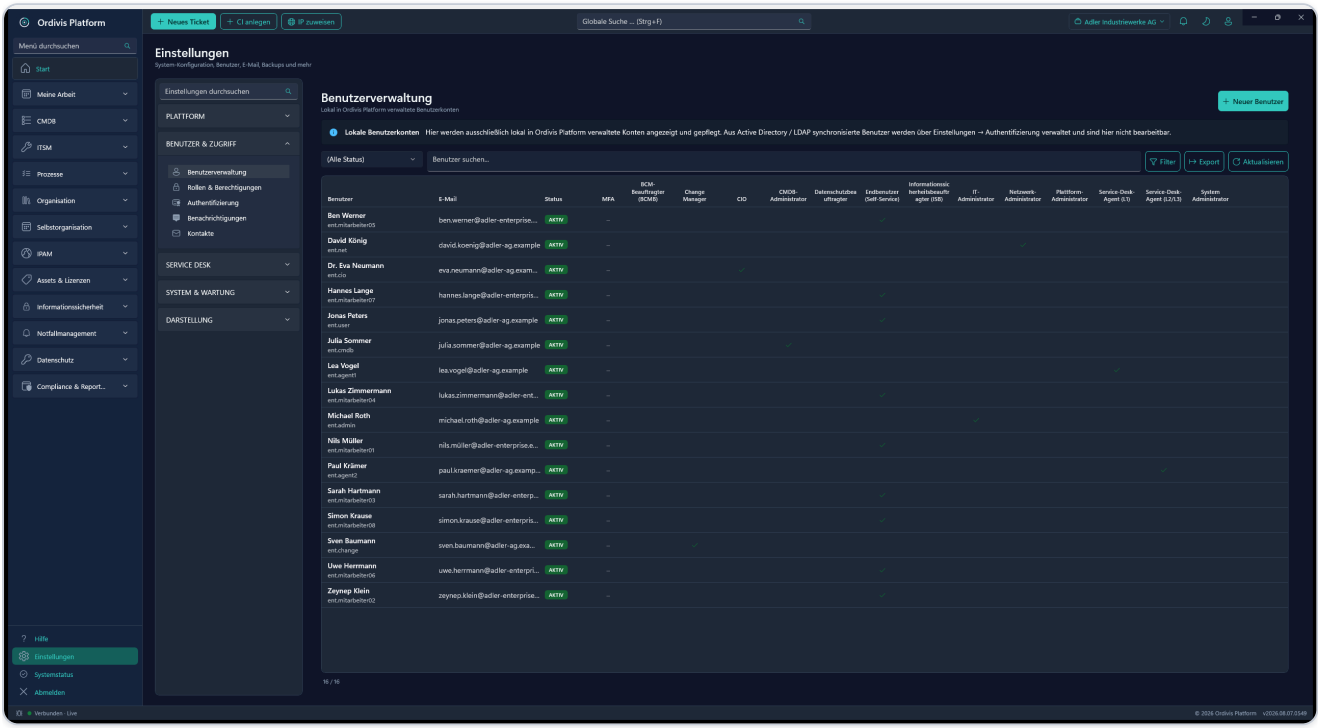

Abb. 2 – Benutzerverwaltung: je Benutzer die zugewiesenen Rollen als angehakte Spalten (Rollen-Matrix).

Beispiele wichtiger Rechte:

Recht	Bedeutung
`system.config.manage`	Administrator (Systemkonfiguration)
`identity.users.manage` / `identity.roles.manage`	Benutzer bzw. Rollen verwalten
`identity.permissions.manage`	Berechtigungen konfigurieren
`tickets.read.all`	alle Tickets (nicht nur Gruppe) sehen
`audit.read` / `audit.export`	Audit-Log lesen / exportieren
`system.security.manage`	Anmeldedaten/Sicherheit verwalten

6 Mehr-Faktor-Authentifizierung (MFA)

MFA ist **freiwillig**: Jeder Benutzer richtet sie für sein eigenes Konto selbst ein – dafür ist kein gesondertes Recht nötig und keines kann sie verhindern. Das **Zurücksetzen fremder** Geräte ist davon getrennt und

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

verlangt `identity.users.manage`.

6.1 Einrichten (Selbstbedienung)

> 💻 **Nur im Client:** Kopfleiste ▸ **Konto-Menü** (Personen-Symbol) ▸ *Zwei-Faktor-Authentifizierung* · 🔑 kein Recht nötig – jeder richtet seinen eigenen Faktor ein

🔒 **Pflichtreihenfolge — erst der bestätigte Code schaltet den zweiten Faktor scharf.** Brechen Sie den Dialog nach Schritt 1 ab, bleibt Ihr Konto unverändert – ein abgebrochener Versuch sperrt Sie also nicht aus.

1. Der Dialog zeigt einen **QR-Code** und daneben das **Base32-Geheimnis** im Klartext. Scannen Sie den Code mit einer Authenticator-App (Microsoft Authenticator, Google Authenticator, Aegis o. ä.) – oder tippen Sie das Geheimnis manuell ein, wenn Sie nicht scannen können.
2. Geben Sie den in der App angezeigten **6-stelligen Code** ein und bestätigen Sie mit *Aktivieren*.

✅ **Ergebnis:** Ab jetzt fragt die Anmeldung nach dem Kennwort zusätzlich nach dem Code – im Client wie im Portal.

Dasselbe Menü schaltet MFA auch wieder ab; der Eintrag wechselt je nach Stand.

Administratoren sehen den MFA-Stand eines Benutzers in der **Benutzerverwaltung** und können den zweiten Faktor dort **zurücksetzen** (etwa bei einem verlorenen Gerät). Danach ist das Konto wieder ohne zweiten Faktor erreichbar, bis der Benutzer ihn neu einrichtet.

> **Hinweis:** Ordivis Platform unterstützt als zweiten Faktor ausschließlich **TOTP** (zeitbasierte Codes, RFC 6238). FIDO2/Hardware-Schlüssel und E-Mail-Einmalcodes sind derzeit **nicht** verfügbar.

6.2 Anmelden mit dem zweiten Faktor

> 💻 **Client:** Anmeldeformular · 🌐 **Portal:** Anmeldeseite · 🔑 kein Recht nötig

Ist der zweite Faktor scharf, läuft die Anmeldung in **zwei Schritten** – das Kennwort allein genügt nicht mehr.

🔒 **Pflichtreihenfolge** — das Code-Feld erscheint erst, nachdem Kennwort und Benutzername abgeschickt wurden. Vorher gibt es nichts einzutragen.

1. Geben Sie **Benutzername** und **Kennwort** ein und schicken Sie ab.
2. Lesen Sie den **sechsstelligen Code** in Ihrer Authenticator-App ab.
3. Tragen Sie ihn in das Feld *Bestätigungscode* ein.
4. **Im Portal zusätzlich:** Geben Sie Benutzername und Kennwort **erneut** mit ein. Das Portal speichert das Kennwort zwischen den Schritten bewusst nicht zwischen.
5. Schicken Sie ab.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

✅ **Ergebnis:** Sie sind angemeldet. Im Client bleiben Kennwort und Mandant nach einem falschen Code stehen; gelöscht wird nur der Code, damit der nächste Versuch ein Klick ist. Die Android-App verhält sich wie der Client.

Der Code ist **30 Sekunden** gültig; die Anwendung akzeptiert zusätzlich den unmittelbar vorangehenden und folgenden Zeitschritt, damit eine leicht abweichende Uhr nicht aussperrt. Ein falscher Code zählt als **Fehlversuch** und läuft in dieselbe Sperre wie ein falsches Kennwort.

> **Hinweis:** Der zweite Faktor tritt **neben** das Kennwort, nicht an seine Stelle. Ein gültiger Code mit falschem Kennwort wird abgewiesen.

6.3 Abschalten verlangt eine Bestätigung

Wer MFA aktiviert hat, muss das Abschalten **erneut bestätigen** – mit dem aktuellen Kennwort **oder** einem gültigen Code aus der Authenticator-App. Ohne Nachweis wird das Abschalten abgelehnt.

Warum: Sonst genügt eine übernommene Sitzung, um den zweiten Faktor still und dauerhaft zu entfernen – der Schutz wäre nur so stark wie das Session-Token.

Beide Wege sind nötig, nicht nur Komfort: Konten aus einem externen Verzeichnis (AD/SAML) haben kein lokales Kennwort und können sich ausschließlich per Code ausweisen.

> **Hinweis:** Solange die Einrichtung **nicht abgeschlossen** ist (kein bestätigtes Gerät), lässt sich der Vorgang ohne Nachweis abbrechen. Es gibt dann noch keinen zweiten Faktor zu schützen – und ein Konto ohne lokales Kennwort käme sonst weder per Kennwort noch per Code an ein Gerät heran, das es nie bestätigt hat.

7 Administrator-Limit (Lizenz)

Die Zahl der **Administratoren** (Benutzer mit `system.config.manage`) kann durch die Produktlizenz begrenzt sein.

> **Achtung:** Ist das Limit erreicht, wird das Anlegen eines Admins bzw. das Zuweisen einer Admin-Rolle mit **HTTP 402** abgelehnt. Siehe ID-43.

8 Verzeichnisbenutzer (AD/LDAP)

Über AD/LDAP synchronisierte Konten (read-only, ID-40) erhalten ihre Rollen ebenfalls hier. Der Sync ändert nie Daten im AD.

8.1 Welche Verzeichnisattribute übernommen werden

Übernommen wird jedes **Textattribut** des Kontos – die benannten Felder (Anzeigename, E-Mail, Abteilung, Telefon, Vorname, Nachname, Funktion) und zusätzlich der vollständige Rohsatz, aus dem sich die Benutzerkonto-CI speist (siehe Kapitel 21b).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Binäre Attribute werden übersprungen – Fotos, Zertifikate, Sicherheitskennungen. Sie tragen keinen lesbaren Text, und ihr Inhalt lässt sich nicht in einem Textfeld ablegen. Fehlt also ein Attribut im Verzeichnisabgleich, das im AD sichtbar ist, ist es mit hoher Wahrscheinlichkeit ein binäres.

8.2 Wie aus AD-Gruppen Rollen werden

In den Systemeinstellungen ordnen Sie AD-Gruppen Rollen zu (ID-40). Jeder Abgleich trägt die passenden Rollen nach.

Die Mitgliedschaft wird über alle Verschachtelungsebenen aufgelöst. Ordnen Sie die Gruppe eines Fachbereichs zu, erhalten auch die Personen ihre Rolle, die nur über eine Fachdienst-Gruppe darin hängen – und ebenso über weitere Zwischenstufen. Das ist der Normalfall in einer gewachsenen Verwaltung: Zugeordnet wird gewöhnlich die obere Gruppe, und darin liegen nicht die Personen, sondern wieder Gruppen. Die Auflösung rechnet der Verzeichnisserver selbst; für Ordivis Platform ist es eine Abfrage je zugeordneter Gruppe, unabhängig von der Zahl der Benutzer.

> **Zuweisen, nicht abgleichen:** Der Lauf **vergibt** Rollen, er nimmt keine weg. Entfernen Sie eine Zuordnung, behalten bereits versorgte Konten ihre Rolle – entziehen müssen Sie sie in der Benutzerverwaltung. So kann ein Fehlgriff in der Zuordnung niemanden aussperren.

Nach dem Ändern einer Zuordnung ist ein Abgleich nötig, damit die Rollen nachgetragen werden; rückwirkend geschieht das nicht von selbst. Wer sofort weiterarbeiten muss, bekommt die Rolle direkt zugewiesen – sie wirkt binnen fünf Minuten.

Was der Lauf meldet. Das Protokoll nennt angelegte und aktualisierte Konten, angetroffene Gruppen und vergebene Rollen. Dabei zählt „Rollen zugewiesen" nur die **neuen** – eine `0` bedeutet also nicht, dass niemand eine Rolle hat, sondern dass sich nichts geändert hat. Gewarnt wird, wenn

- keine Zuordnung hinterlegt ist,
- eine zugeordnete Gruppe bei **keinem** Benutzer angetroffen wurde (umbenannt? verschoben?), oder
- **mehr** Benutzer ohne Rolle dastehen als beim vorigen Lauf.

Der reine Bestand an Konten ohne Rolle ist bewusst **keine** Warnung: Dienst- und Funktionskonten sind zu Recht rollenlos, in größeren Verwaltungen mehrere Hundert. Eine Warnung, die bei jedem Lauf erscheint, wird nicht mehr gelesen – und der eine Lauf, an dem wirklich etwas kippt, ginge darin unter. Den Bestand sehen Sie in der Verzeichnis-Ansicht.

> **Ein Konto ohne Rolle hat kein einziges Recht.** Die Anmeldung gelingt, danach wird jeder Aufruf abgewiesen; im Self-Service-Portal meldet „Meine Vorgänge" nur, dass sich nichts laden lässt. Wer das als Ausfall gemeldet bekommt, prüft zuerst die Rollenzuordnung.

Verwandte Themen

- **ID-44 – Rollen- & Rechte-Matrix** – der vollständige Rechtekatalog
- **ID-40 – Administrationshandbuch** – Ersteinrichtung und Systemeinstellungen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-42 – Modul Mandantenfähigkeit** – Benutzer in mehreren Mandanten
- **ID-21b – Benutzerkonten in der CMDB** – dieselben Konten als Configuration Item

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · MANDANTENFÄHIGKEIT

Modul Mandantenfähigkeit (Multi-Tenancy)

Mehrere Organisationen in einer Installation – Trennung und Verwaltung

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-42 Mandanten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Mandantenfähigkeit (Multi-Tenancy)

Ein IT-Dienstleister betreut zehn Kunden; ein Konzern hat mehrere Tochtergesellschaften; eine Stadtverwaltung trennt ihre Ämter. In all diesen Fällen sollen mehrere Organisationen dieselbe Software nutzen, ohne dass die eine die Daten der anderen sieht. Für jede Organisation eine eigene Installation zu betreiben, wäre teuer und in der Pflege aufwendig.

Mandantenfähigkeit (englisch *Multi-Tenancy*) löst das. Ein Mandant ist eine abgegrenzte Organisation innerhalb derselben Installation. Alle arbeiten auf demselben System, aber jeder sieht ausschließlich seine eigenen Tickets, CIs, Assets und Benutzer. Man kann sich das wie ein Bürogebäude mit vielen Mietern vorstellen: gemeinsame Technik im Haus, aber abgeschlossene Einheiten. Dieses Modul beschreibt, wie Mandanten angelegt, getrennt und verwaltet werden.

INHALT

1 Was ist ein Mandant?
2 Mandanten anlegen & verwalten
3 Datentrennung
4 Zentrale Identität & Umschalten
5 Hierarchie & Roll-up
6 Cross-Tenant-Zugriff (Grants)
7 Architektur-Ausblick
8 Rollen & Rechte (Auszug)
Verwandte Themen

1 Was ist ein Mandant?

Ein **Mandant** ist eine abgegrenzte Organisationseinheit. Tickets, CIs, Assets, Benutzer und Einstellungen gehören jeweils zu einem Mandanten. Ein Benutzer kann mehreren Mandanten zugeordnet sein und beim Login den Mandanten wählen.

2 Mandanten anlegen & verwalten

💻 **Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Mandanten* · 🌐 **Portal:** *Verwaltung* ▸ *Mandanten und Module* · 🔑 `tenant.manage`

🔒 **Pflichtreihenfolge** — ein Benutzer lässt sich erst zuweisen, wenn der Mandant existiert, und die Modulschalter greifen erst danach. Die Kennung aus Schritt 2 ist **nicht mehr änderbar**, sobald Daten im Mandanten liegen.

1. Öffnen Sie die Mandantenverwaltung.
2. Legen Sie einen Mandanten an: *Name*, *Kennung* und *Status*.
3. Weisen Sie **Benutzer** zu. Das Verhältnis ist n:m – ein Benutzer kann zu mehreren Mandanten gehören.
4. Schalten Sie die benötigten **Module** ein oder aus.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5. Aktivieren oder deaktivieren Sie den Mandanten nach Bedarf.

✅ **Ergebnis:** Der Mandant steht bereit. Benutzer mit mehreren Mandanten bekommen bei der Anmeldung die Auswahl (ID-01, Abschnitt 3.1).

> **Achtung:** Ein deaktivierter Mandant ist nicht gelöscht – seine Daten bleiben vollständig erhalten und zählen weiter gegen die Lizenzgrenzen (ID-43).

3 Datentrennung

Die Trennung erfolgt serverseitig über **Mandantenfilter** in jeder Datenbankabfrage. Anwender sehen ausschließlich Daten ihres aktiven Mandanten.

> **Achtung (Betrieb):** Der Mandantenfilter muss den **aktuellen** Mandantenkontext live referenzieren. Wird die Mandanten-ID einmalig in das Modell eingebacken, werden Daten unsichtbar. Dies ist ein bekannter Implementierungshinweis für Entwickler/Betrieb.

4 Zentrale Identität & Umschalten

Über eine **Registry-Datenbank** werden Mandanten zentral registriert. Berechtigte Benutzer können zwischen Mandanten **umschalten**, ohne sich neu anzumelden.

5 Hierarchie & Roll-up

Mandanten können in einer **Hierarchie** stehen. Über einen **Roll-up** lassen sich Kennzahlen über einen Mandanten-Teilbaum aggregieren (z. B. Konzernsicht über Tochtergesellschaften).

6 Cross-Tenant-Zugriff (Grants)

Cross-Tenant Access Grants erlauben kontrollierten, zeitlich begrenzten Zugriff über Mandantengrenzen hinweg (z. B. zentrale IT unterstützt einen Tochter-Mandanten) – nachvollziehbar und entziehbar.

7 Architektur-Ausblick

Zielbild ist **eine eigene Datenbank je Mandant** plus zentrale Registry. Der aktuelle Stand und die konkrete Ausbaustufe hängen von der installierten Version ab; das Backend (Registry, Provisioning, zentrale Identität, Umschalten, Grants) ist vorhanden.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

8 Rollen & Rechte (Auszug)

Recht	Erlaubt
`tenant.manage`	Mandanten anlegen, archivieren, löschen; Hierarchie und Zuordnungen
`tenant.grant.manage`	Mandantenübergreifende Zugriffe verwalten
`tenant.rollup.read`	Mandantenübergreifende Auswertung lesen
(kein eigenes Recht)	Das **Umschalten** zwischen den eigenen Mandanten steht jeder angemeldeten Person offen; `tenant.manage` erweitert die Auswahl auf alle Mandanten.

Verwandte Themen

- **ID-41 – Modul Identität, Benutzer & Rollen** – Benutzer und Rollen je Mandant
- **ID-43 – Modul Produktlizenzierung** – Mandantenzahl und Limits der Lizenz
- **ID-40 – Administrationshandbuch** – Verwaltung im Administrationshandbuch
- **ID-60 – API-Referenz** – Mandantenauswahl an der Schnittstelle

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

MODUL · PRODUKTLIZENZIERUNG

Modul Produktlizenzierung & Aktivierung

Testphase, Aktivierung, Editionen und Limits von Ordivis Platform selbst

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-43 Lizenzierung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Modul Produktlizenzierung & Aktivierung

Wie fast jede professionelle Software wird auch Ordivis Platform lizenziert: Ein gültiger Lizenzschlüssel schaltet das Produkt frei und legt fest, in welchem Umfang Sie es nutzen dürfen. Dieses Dokument erklärt, wie diese Produktlizenz funktioniert, vom kostenlosen Testzeitraum über die Aktivierung bis zu dem, was passiert, wenn eine Lizenz ausläuft.

Eine Verwechslung sei gleich zu Beginn ausgeräumt: Hier geht es um die Lizenz *für Ordivis Platform selbst*, also um Ihr Recht, diese Software einzusetzen. Nicht gemeint sind die Lizenzen der Programme, die Ihre Organisation verwaltet (etwa Office- oder Fachanwendungen). Diese kundeneigenen Software-Lizenzen gehören ins Asset-Modul (ID-23).

INHALT

1 Testphase
2 Lizenz importieren & aktivieren
Hardware-Fingerprint & statische MAC-Adresse (Voraussetzung)
3 Editionen & Limits
4 Verhalten nach Ablauf
5 Lizenzstatus im Blick
6 Rollen & Rechte (Auszug)
Verwandte Themen

1 Testphase

Nach der Installation läuft eine **Testphase von 30 Tagen** mit **vollem Funktionsumfang**. Der Lizenzstatus und die verbleibenden Tage werden im Client angezeigt.

2 Lizenz importieren & aktivieren

Abbildung 1 zeigt die Seite *Lizenz & Aktivierung* mit allen genannten Bereichen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

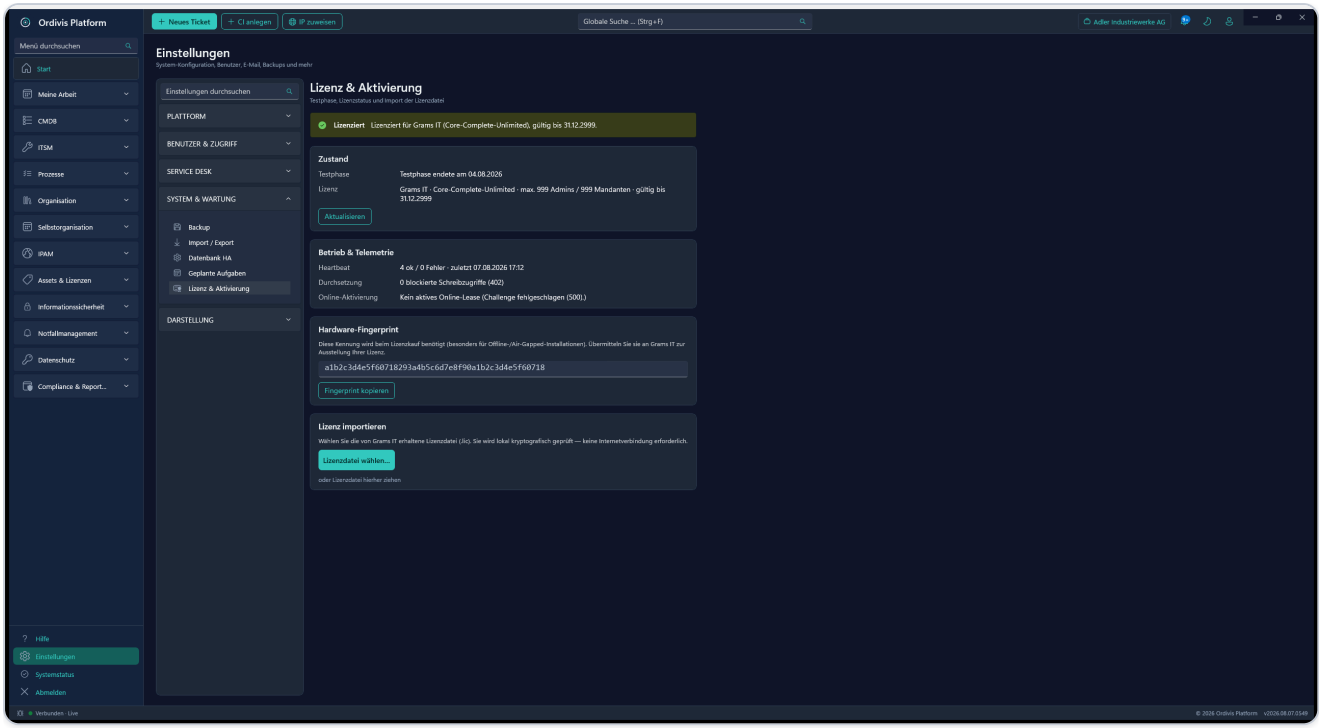

Abb. 1 – Lizenz & Aktivierung: Lizenzstatus, Edition/Limits, Betriebs-Telemetrie, Hardware-Fingerprint und Import der Lizenzdatei.

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Lizenz & Aktivierung* · 🔑 `licensing.manage` — Das Portal zeigt einen Hinweis, wenn die Lizenz abläuft, führt die Aktivierung aber nicht durch.

> **Voraussetzung:** Die signierte Lizenzdatei liegt vor. Sie ist an den **Fingerprint** dieses Systems gebunden – eine Datei für einen anderen Server wird abgelehnt.

🔒 **Pflichtreihenfolge** — der Fingerprint muss dem Herausgeber **vor** der Ausstellung vorliegen. Wer die Datei zuerst anfordert und den Fingerprint danach abliest, bekommt eine Lizenz, die nicht passt.

1. Öffnen Sie *Einstellungen* ▸ *Lizenz & Aktivierung* und lesen Sie den **Hardware-Fingerprint** ab.
2. Fordern Sie beim Herausgeber die Lizenzdatei für genau diesen Fingerprint an.
3. Importieren Sie die **signierte Lizenzdatei**.
4. Ordivis Platform prüft die Signatur (**ECDSA**) und bindet die Lizenz an den Fingerprint.

✅ **Ergebnis:** Der Lizenzstatus wechselt auf **aktiv**; Edition und Limits werden angezeigt (Abbildung 1).

Offline- vs. Online-Aktivierung: Die Aktivierung funktioniert **immer offline** über die signierte Lizenzdatei – eine Internet-Verbindung ist für den Betrieb nicht erforderlich. Ist die **Online-Aktivierung** aktiviert (Standard bei Neuinstallationen), meldet sich Ordivis Platform zusätzlich regelmäßig beim Aktivierungsdienst und holt eine kurzlebige **Bestätigung (Lease)**. Der Lizenzstatus zeigt dann **Online-Aktivierung: ja** an. Diese Online-Prüfung ist **beratend**: Ist der Dienst nicht erreichbar, bleibt die Installation über die Lizenzdatei uneingeschränkt gültig – es entsteht **keine Sperre** durch fehlende Netz-Verbindung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> **Datenschutz:** Übertragen werden ausschließlich Lizenz-Kennung und ein anonymer Hardware-Fingerprint (ein Hash, keine personenbezogenen Daten). Wer die Online-Aktivierung nicht wünscht, deaktiviert sie in `appsettings.json` (`Activation:OnlineMode = false`) und betreibt Ordivis Platform rein offline.

Hardware-Fingerprint & statische MAC-Adresse (Voraussetzung)

Der Fingerprint wird aus **stabilen** Hardware-Merkmalen gebildet: der Windows-`MachineGuid` und den MAC-Adressen der **physischen** Netzwerkkarten. Bewusst **nicht** einbezogen werden reine Host-/Virtualisierungs-Adapter (Hyper-V Default Switch, VPN-/TAP-Tunnel, WSL-/Docker-Bridges) – diese dürfen wechseln, ohne die Aktivierung zu beeinflussen.

> **Voraussetzung in virtuellen Maschinen:** Die MAC-Adresse der **primären vNIC muss statisch** sein. Konfigurieren Sie in Ihrem Hypervisor eine **feste MAC** (Hyper-V/VMware/Proxmox: „static" statt „dynamic"). Ändert sich die MAC beim Neustart oder bei einer Live-Migration, ändert sich der Fingerprint und die Aktivierung sperrt fälschlich mit **„Manipulation erkannt"**.

Wechselt die Hardware **legitim** (z. B. NIC-Tausch, geplante Migration), erkennt Ordivis Platform dies an den untereinander konsistenten, aber vom aktuellen System abweichenden Aktivierungs-Ankern und **bindet automatisch neu** – ohne Datenverlust und ohne dass die Testphase zurückgesetzt wird. Bei einer an eine **bestimmte** Hardware gebundenen (Single-Node-)Lizenz fordern Sie nach einem Hardware-Wechsel eine **Re-Aktivierung** beim Herausgeber an.

3 Editionen & Limits

Die Lizenz legt fest:

- **maximale Administratoren** (Benutzer mit `system.config.manage`),
- **maximale Mandanten**,
- **freigeschaltete Module**.

> **Achtung:** Das Überschreiten des **Administrator-Limits** wird beim Anlegen bzw. Zuweisen einer Admin-Rolle mit **HTTP 402** abgelehnt (siehe ID-41).

4 Verhalten nach Ablauf

Läuft die Lizenz ab, wechselt Ordivis Platform in einen **eingeschränkten Modus**: **schreibende** API-Aufrufe werden mit **HTTP 402** abgewiesen, lesender Zugriff bleibt möglich, damit keine Daten verloren gehen. Im Portal und Client erscheint ein Hinweis-Banner.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5 Lizenzstatus im Blick

Ein **Banner** und die Lizenzseite zeigen jederzeit Status, Edition, Resttage und Limits. Läuft die Lizenz bald ab, werden Administratoren rechtzeitig benachrichtigt.

6 Rollen & Rechte (Auszug)

Recht	Erlaubt
`licensing.read`	Lizenzstatus ansehen
`licensing.manage`	Lizenz importieren und aktivieren

Verwandte Themen

- **ID-42 – Modul Mandantenfähigkeit** – Limits je Mandant
- **ID-41 – Modul Identität, Benutzer & Rollen** – gezählte Benutzer und Rollen
- **ID-23 – Modul Asset- & Vertragsmanagement** – Abgrenzung zu Softwarelizenzen Ihrer eigenen Programme
- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Aktivierung auf dem Server

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TECHNISCHE REFERENZ

Rollen- & Rechte-Matrix (RBAC)

RBAC – Rollen, Berechtigungen und vollständige Rechte-Matrix

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-44 Rollen & Rechte

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Rollen- & Rechte-Matrix (RBAC)

Technisches Referenzdokument für Fachpersonal (Administratoren, Auditoren, Sicherheitsverantwortliche). Es dokumentiert das rollenbasierte Berechtigungsmodell von Ordivis Platform vollständig: das Berechtigungskonzept, alle **Systemrollen**, den **Berechtigungskatalog** sowie die exakte Zuordnung Rolle → Berechtigungen.

Hinweis: Grundlagen zur Bedienung der Benutzerverwaltung (Benutzer anlegen, MFA, Kennwort) stehen im Modul-Dokument **ID-41**. Dieses Dokument ist die technische Referenz zum Rechtemodell.

INHALT

1 Berechtigungskonzept
- 1.1 SuperAdmin und die Plattform-Ebene

2 Systemrollen

3 Berechtigungskatalog
- 3.1 CMDB (cmdb)
- 3.2 Ticketing / Service Desk (ticketing)
- 3.2a Prozessdokumentation (process)
- 3.3 Wissensdatenbank (knowledge)
- 3.4 IPAM (ipam)
- 3.5 Change (change)
- 3.6 Identität (identity)
- 3.7 System, Audit, Reporting, Lizenz
- 3.8 Verträge, Discovery, Workflow, DR
- 3.9 Informationssicherheit (security)
- 3.10 Business Continuity (bcm)
- 3.11 Datenschutz (datenschutz)
- 3.12 Checklisten (checklist)
- 3.13 Handbücher (handbook)
- 3.14 Organisation und Geschäftsverteilungsplan (gvp)
- 3.15 Plattform-Betrieb (platform)
- 3.16 Maschinen-Zugänge (Integrations-Token)

4 Rolle → Berechtigungen (exakte Zuordnung)

5 Besondere Hinweise
- 5.1 Funktionstrennung (Vier-Augen-Prinzip)
- 5.2 Nach dem Update: einmalige Neuanmeldung erforderlich

6 Bedienung in der Anwendung
- 6.1 Berechtigungsmatrix (Rollen × Berechtigungen)
- 6.2 Mandanten-Verwaltung (nur SuperAdmin / Plattform-Administrator)
- 6.3 Mandanten-Umschalter
- 6.4 DSGVO-Betroffenenrechte

Verwandte Themen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1 Berechtigungskonzept

- **Modell:** Role-Based Access Control (RBAC). Berechtigungen (*Permissions*) hängen an **Rollen**, Rollen an **Benutzern**. Ein Benutzer kann mehrere Rollen besitzen; die effektiven Rechte sind die **Vereinigung** aller Rollenrechte.
- **Token:** Nach der Anmeldung enthält das JWT ausschließlich `permission`**-Claims** – keine Rollennamen. Client und Portal steuern Sichtbarkeit anhand dieser Claims.
- **Durchsetzung (Server):** Jeder geschützte Endpunkt trägt `[RequiresPermission("<key>")]`. Fehlt das Recht, antwortet die API mit **HTTP 403**.
- **Schlüsselschema:** `<modul>.<objekt>.<aktion>` (z. B. `cmdb.ci.update`). Lesen endet meist auf `.read`, verwalten auf `.manage`.
- **Prinzip der minimalen Rechte:** Eine Rolle trägt nur die tatsächlich benötigten Rechte. Wie eine Rolle zusammengestellt wird, steht in Abschnitt 6.1.

1.1 SuperAdmin und die Plattform-Ebene

Ordivis Platform unterscheidet zwei Ebenen:

- **Mandanten-Ebene** – der normale Betrieb *innerhalb* eines Mandanten (CMDB, Ticketing, IPAM …). Hier arbeiten die operativen Rollen, allen voran der **IT-Administrator** (`ItAdmin`).
- **Plattform-Ebene** – die mandantenübergreifende Verwaltung *der Mandanten selbst* (Mandant anlegen, archivieren, löschen, Hierarchie, mandantenübergreifende Zugriffe).

Der **erste, bei der Installation eingerichtete Benutzer** ist der **SuperAdmin**. Er ist bewusst **speziell** und von den gewöhnlichen IT-Systemadministratoren einer IT-Abteilung zu unterscheiden: nur er (bzw. Rollen, denen er die Plattform-Rechte ausdrücklich delegiert) darf neue Mandanten anlegen oder löschen. Technisch trägt der SuperAdmin die Rolle **SystemAdmin** (alle Rechte inklusive der Plattform-Berechtigungen `tenant.*`).

Für die **Vertretung** kann der SuperAdmin einzelnen Benutzern die Rolle **Plattform-Administrator** geben – diese erlaubt ausschließlich die Mandanten-Verwaltung (`tenant.manage` , `tenant.rollup.read`), **nicht** die volle Systemmacht. Ein gewöhnlicher **IT-Administrator** (`ItAdmin`) besitzt **keine** `tenant.*` -Rechte und kann daher keine fremden Mandanten provisionieren oder löschen.

2 Systemrollen

Ordivis Platform liefert 13 vordefinierte **Systemrollen** aus (bei jedem Start konvergent nachgezogen). Eigene Rollen lassen sich zusätzlich anlegen und mit beliebigen Berechtigungen bestücken.

Rolle	Bezeichnung	Kurzbeschreibung
SystemAdmin	System Administrator	Voller Systemzugriff (**alle** Rechte) – die Rolle des **SuperAdmins** (s. 1.1)
PlatformAdmin	Plattform-Administrator	Mandanten anlegen/archivieren/löschen + Roll-up (Vertretung des SuperAdmins, **keine** volle Systemmacht)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Rolle	Bezeichnung	Kurzbeschreibung
ItAdmin	IT-Administrator	Vollzugriff auf alle operativen Module (Ausnahmen s. u.)
SdAgentL1	Service-Desk-Agent (L1)	Tickets bearbeiten, CMDB lesen, KB pflegen
SdAgentL2	Service-Desk-Agent (L2/L3)	Erweiterte Ticket-/CI-/Change-/Workflow-Rechte
CmdbAdmin	CMDB-Administrator	CMDB-Klassen und -Strukturen verwalten
NetworkAdmin	Netzwerk-Administrator	IPAM vollständig verwalten
ChangeManager	Change Manager	Change Requests freigeben/umsetzen
Enduser	Endbenutzer (Self-Service)	Eigene Tickets, Servicekatalog, öffentliche KB
CIO	CIO	Strategische Übersicht, Dashboards, lesend
DSB	Datenschutzbeauftragter	Audit-Log, DSGVO-relevante Leserechte
ISB	Informationssicherheitsbeauftragter	Vollzugriff ISMS/IT-Grundschutz, genehmigt Risiken
BCMB	BCM-Beauftragter	Vollzugriff Business Continuity Management

3 Berechtigungskatalog

Der Katalog umfasst **140 Berechtigungen**, gruppiert nach Modul. `key` ist der technische Wert im JWT und in `[RequiresPermission]`.

3.1 CMDB (`cmdb`)

key	Beschreibung
cmdb.ci.read	CI-Datensätze lesen
cmdb.ci.create	Neue CIs anlegen
cmdb.ci.update	CIs bearbeiten
cmdb.ci.delete	CIs löschen
cmdb.ci.lifecycle.manage	CI-Lifecycle-Status ändern
cmdb.class.manage	CMDB-Klassen definieren
cmdb.relation.manage	CI-Beziehungen verwalten
cmdb.import	Massen-Import
cmdb.export	Export von CI-Listen
cmdb.baseline.manage	Baselines verwalten

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3.2 Ticketing / Service Desk (`ticketing`)

key	Beschreibung
tickets.read	Tickets der eigenen Support-Gruppe(n) lesen
tickets.read.all	Tickets ALLER Support-Gruppen lesen (kein Gruppen-Filter)
tickets.self.read	Nur eigene Tickets lesen
tickets.create	Tickets erstellen
tickets.update	Tickets bearbeiten
tickets.assign	Tickets zuweisen
tickets.close	Tickets schließen
tickets.delete	Tickets löschen
tickets.escalate	Tickets eskalieren
tickets.approve	Service-Request-Freigaben entscheiden
tickets.export	Ticket-Listen exportieren
tickets.sla.manage	SLA-Profile konfigurieren
ticketing.category.manage	Ticket-Kategorien verwalten
ticketing.config.manage	**Neu:** Service-Desk-Konfiguration verwalten – Routing-Regeln, Automatisierungsregeln, Genehmigungsstufen, Support-Gruppen, Wartungsfenster, Textbausteine und Vorlagen
contacts.manage	Lokale Kontakte (externe Melder) verwalten
ticketing.csat.submit	Zufriedenheit (CSAT) abgeben
ticketing.service_catalog.read	Service-Katalog ansehen/bestellen
ticketing.service_catalog.manage	Service-Katalog (Artikel/Kategorien) verwalten

Wichtig (seit 2026.07.14.1845): Die Steuerungsobjekte des Service Desks – SLA-Profile, Routing, Automatisierungsregeln, Genehmigungsstufen, Support-Gruppen, Wartungsfenster und Textbausteine – ließen sich zuvor mit dem reinen Leserecht `tickets.read` ändern. Sie verlangen jetzt `ticketing.config.manage` bzw. `tickets.sla.manage` . Diese Rechte besitzen der **Service-Desk-Agent (L2)** sowie IT- und System-Administratoren; ein **L1-Agent** kann die Konfiguration nun nicht mehr verändern (die zugehörigen Schaltflächen sind für ihn ausgeblendet).

3.2a Prozessdokumentation (`process`)

key	Beschreibung
process.manage	**Neu:** Prozessdokumentation pflegen – Steckbrief, RACI, KPI, Datenobjekte, BPMN-Entwurf

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

key	Beschreibung
process.approve	**Neu:** Prozessdefinition freigeben bzw. verwerfen (IKS-Kontrolle)

Funktionstrennung: Pflege und Freigabe sind bewusst **getrennte** Rechte. Der **CMDB-Administrator** pflegt die Prozessdokumentation (`process.manage`), die Freigabe (`process.approve`) liegt bei der Leitungsebene (**CIO**). Damit ist die Freigabe einer Prozessdefinition eine eigenständige Kontrollhandlung im Sinne des internen Kontrollsystems (ISO 9001, ISO 27001 A.5.37) und nicht mehr durch jeden CMDB-Leser vollziehbar.

3.3 Wissensdatenbank (`knowledge`)

key	Beschreibung
knowledge.read.public	Öffentliche KB-Artikel lesen
knowledge.read.internal	Interne KB-Artikel lesen
knowledge.author	KB-Artikel erstellen/bearbeiten
knowledge.review	KB-Artikel prüfen (Review-Gate)
knowledge.publish	KB-Artikel veröffentlichen/zurückziehen
knowledge.admin	KB-Artikel löschen/verwalten

3.4 IPAM (`ipam`)

key	Beschreibung
ipam.prefix.read	Subnetze lesen
ipam.prefix.manage	Subnetze verwalten
ipam.ip.read	IP-Adressen lesen
ipam.ip.assign	IPs reservieren/zuweisen
ipam.ip.free	IPs freigeben
ipam.ip.confirm	Unbestätigte IPs bestätigen
ipam.vrf.manage	VRFs verwalten
ipam.dhcp.manage	DHCP-Konfiguration verwalten
ipam.dhcp.import	Windows-DHCP-Server importieren
ipam.scan.trigger	IP-Scan auslösen
ipam.export	IP-Listen exportieren

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3.5 Change (`change`)

key	Beschreibung
change.read	Change Requests lesen
change.create	Change Requests erstellen
change.approve	Change Requests genehmigen
change.reject	Change Requests ablehnen
change.implement	Change als umgesetzt markieren
change.emergency	Emergency Changes erstellen

3.6 Identität (`identity`)

key	Beschreibung
identity.users.read	Benutzerliste lesen
identity.users.manage	Benutzer verwalten
identity.roles.read	Rollenkonfiguration lesen
identity.roles.manage	Rollen verwalten
identity.permissions.manage	Berechtigungen konfigurieren
identity.mfa.enforce	MFA-Pflicht setzen
identity.sessions.manage	Aktive Sessions verwalten

3.7 System, Audit, Reporting, Lizenz

key	Modul	Beschreibung
system.config.read	system	Systemkonfiguration lesen
system.config.manage	system	Systemkonfiguration ändern (= „**Administrator**“)
system.backup.manage	system	Backups verwalten
system.security.manage	system	Sicherheitseinstellungen
reporting.read	reporting	Berichte lesen
reporting.create	reporting	Berichte erstellen – **und** die Vorschau freier SQL-Abfragen (freies SQL läuft an den Modul-Berechtigungen vorbei, siehe Kapitel *Reports & SQL-Abfragen*)
reporting.manage	reporting	Alle Berichte verwalten
reporting.dashboard.manage	reporting	Dashboard-Widgets konfigurieren
audit.read	audit	Audit-Log lesen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

key	Modul	Beschreibung
audit.export	audit	Audit-Log als CSV ausgeben (zusätzlich zu `audit.read`; ohne dieses Recht wird der Abruf abgewiesen)
licensing.read	licensing	Lizenzdaten lesen
licensing.manage	licensing	Lizenzen verwalten
activation.manage	activation	Produktaktivierung / Lizenz verwalten
notifications.mailbox.admin	notifications	E-Mail-Postfächer (Versand + Ticket) verwalten
notifications.messenger.admin	notifications	Messenger-Verbindungen (Mattermost, Webhook) verwalten
discovery.action1.admin	discovery	Action1-Anbindung verwalten (Zugang, Organisation, Datenquellen)
tenant.manage	platform	**Plattform:** Mandanten anlegen/archivieren/löschen/Hierarchie (nur SuperAdmin / Plattform-Administrator)
tenant.rollup.read	platform	**Plattform:** mandantenübergreifendes Roll-up-Aggregat lesen
tenant.grant.manage	platform	**Plattform:** mandantenübergreifende Zugriffs-Grants verwalten (nur SuperAdmin)
privacy.manage	privacy	DSGVO-Betroffenenrechte (Auskunft/Löschung) + Verarbeitungsverzeichnis (DSB)

Die `tenant.*`-Berechtigungen bilden die **Plattform-Ebene** (s. 1.1). Sie sind bewusst von der mandantenweiten `system.config.manage` getrennt: Ein IT-Administrator besitzt Letztere, aber **keine** `tenant.*`-Rechte und kann somit keine fremden Mandanten verwalten.

3.8 Verträge, Discovery, Workflow, DR

key	Modul	Beschreibung
contracts.read	contracts	Verträge und Lieferanten lesen
contracts.manage	contracts	Verträge/Lieferanten/Verpflichtungen verwalten
discovery.read	discovery	Discovery-/Scan-Daten lesen
discovery.manage	discovery	Scans auslösen, Anmeldedaten/Anhänge verwalten
discovery.collector.manage	discovery	Discovery-Collector anlegen, anmelden, sperren
discovery.result.review	discovery	Gefundene Clients prüfen und als CI übernehmen
workflow.read	workflow	Workflows und Aufgaben lesen
workflow.manage	workflow	Workflows starten und Aufgaben bearbeiten
dr.read	dr	Notfall-/DR-Pläne lesen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

key	Modul	Beschreibung
dr.manage	dr	Notfall-/DR-Pläne und Testergebnisse verwalten

3.9 Informationssicherheit (`security`)

key	Beschreibung
security.catalog.manage	Sicherheitskataloge (BSI/ISO) importieren/verwalten
security.scope.manage	Informationsverbund/Geltungsbereich verwalten
security.assessment.manage	Schutzbedarfsfeststellung verwalten
security.modelling.manage	Modellierung (Baustein-Zuweisung) verwalten
security.audit.manage	IT-Grundschutz-Checks/Audits verwalten
security.check.write	Prüfergebnisse erfassen
security.check.read	ISMS-Daten/Prüfergebnisse lesen
security.risk.manage	Risikoanalysen verwalten
security.risk.accept	Risiken freigeben/akzeptieren
security.report.generate	ISMS-Berichte erzeugen
security.report.read	ISMS-Berichte/Dashboards lesen
security.measures.own.write	Eigene zugewiesene Maßnahmen bearbeiten

3.10 Business Continuity (`bcm`)

key	Beschreibung
bcm.scope.manage	BCM-Geltungsbereich/Verbund verwalten
bcm.process.manage	BCM-Prozess-Overlay verwalten
bcm.bia.manage	Business Impact Analyse (BIA) verwalten
bcm.resource.link	Ressourcen (CI/Vendor/Person) verknüpfen
bcm.risk.read	BCM-Risiko-/Gap-Daten lesen
bcm.org.manage	Bewältigungsorganisation (AAO/BAO) verwalten
bcm.plan.manage	Notfallpläne/Checklisten verwalten
bcm.plan.read	Notfallpläne/Checklisten lesen (Portal)
bcm.exercise.manage	Übungen/Tests verwalten
bcm.capa.manage	Korrektur-/Verbesserungsmaßnahmen (CAPA) verwalten
bcm.report.generate	BCM-Berichte erzeugen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

key	Beschreibung
bcm.report.read	BCM-Berichte/Dashboards lesen
bcm.crisis.operate	Krisenmodus/BAO-Aktivierung bedienen

3.11 Datenschutz (`datenschutz`)

Das Verzeichnis der Verarbeitungstätigkeiten (VVT) trennt bewusst zwischen **eigenen** und **allen** Verfahren: `datenschutz.vvt.read` zeigt die Verfahren des eigenen Bereichs, erst `datenschutz.vvt.read.all` das gesamte Verzeichnis – dieses Recht gehört der oder dem Datenschutzbeauftragten und dem Betriebsrat, nicht jeder fachlich beteiligten Person.

key	Beschreibung
datenschutz.vvt.read	Verzeichnis lesen (Standard-Umfang: eigene Verfahren)
datenschutz.vvt.read.all	Gesamtes Verzeichnis lesen (DSB/Betriebsrat)
datenschutz.vvt.edit	Verarbeitungstätigkeiten anlegen/ändern
datenschutz.vvt.approve	Verarbeitungstätigkeiten freigeben (DSB)
datenschutz.vvt.export	Behördenkonformen Export erzeugen
datenschutz.catalog.manage	Kataloge (Betroffene/Datenkategorien) verwalten
datenschutz.breach.read	Datenschutzverletzungen lesen (Art. 33/34 DSGVO)
datenschutz.breach.report	Datenschutzverletzung melden/erfassen
datenschutz.breach.manage	Datenschutzverletzungen bewerten/melden/abschließen (DSB)

3.12 Checklisten (`checklist`)

`checklist.run.execute.self` ist bewusst eng: Damit arbeitet jemand **die ihm zugewiesene** Durchführung ab, ohne fremde Durchführungen zu sehen oder starten zu können.

key	Beschreibung
checklist.read	Checklisten und Durchführungen lesen
checklist.template.manage	Checklisten-Vorlagen pflegen
checklist.run.manage	Durchführungen starten/zuweisen/abbrechen
checklist.run.execute.self	Eigene zugewiesene Checkliste abarbeiten

3.13 Handbücher (`handbook`)

key	Beschreibung
handbook.read	Handbücher lesen (Betrieb/Notfall/Wiederanlauf)
handbook.manage	Handbücher pflegen/generieren (Kapitel, Bausteine, Versionierung)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

key	Beschreibung
handbook.approve	Handbuch freigeben (Entwurf → freigegebener Stand)

3.14 Organisation und Geschäftsverteilungsplan (`gvp`)

key	Beschreibung
gvp.read	Produktkatalog, Ablauf-/Aufbauorganisation und Geschäftsverteilungsplan lesen
gvp.manage	Katalogpflege, Übernahme, Zuständigkeiten und GVP-Fassungen verwalten

3.15 Plattform-Betrieb (`platform`)

Wie die `tenant.*`-Rechte gehören diese der **Plattform-Ebene** (s. 1.1) und nicht in die Rolle eines mandantenbezogenen Administrators.

key	Beschreibung
platform.monitor	Plattform-Überwachung lesen (PostgreSQL, Sitzungen, Betriebskennzahlen der Mandanten)
platform.hosting.manage	HTTPS/Transportsicherheit konfigurieren (Zertifikat, ACME, Reverse-Proxy)
platform.eventlog.manage	Windows-Sicherheitsprotokoll (Ereignisanzeige) konfigurieren
platform.integrationtoken.manage	Integrations-Token (Maschinen-Zugang) ausstellen und widerrufen
system.backup.restore	Datenbank aus einer Sicherung wiederherstellen (**destruktiv**)

> `system.backup.restore` überschreibt den laufenden Datenbestand. Es gehört keiner Rolle, die im Tagesgeschäft benutzt wird, sondern wird für den Wiederherstellungsfall gezielt vergeben.

3.16 Maschinen-Zugänge (Integrations-Token)

Diese Berechtigungen stehen im Katalog, gehören aber **nicht in eine Benutzerrolle**. Sie werden ausschließlich von **Integrations-Token** getragen (externe Werkzeuge, Discovery-Collector). Die zugehörigen Endpunkte akzeptieren nur das Token-Verfahren – ein angemeldeter Benutzer erreicht sie auch mit dem passenden Recht nicht.

key	Beschreibung
ipam.scan.ingest	Scan-Ergebnisse externer Agenten einspeisen
cmdb.ingest	Geräte und Zertifikate externer Agenten als Funde einliefern
security.finding.ingest	Schwachstellenbefunde externer Agenten einliefern
discovery.collector.report	Collector meldet Auftrag, Lebenszeichen und Funde

`discovery.collector.report` lässt sich **nicht von Hand ausstellen**: Diesen Token vergibt ausschließlich das Enrollment eines Collectors, gebunden an genau einen Collector.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4 Rolle → Berechtigungen (exakte Zuordnung)

Die folgende Auflistung entspricht der ausgelieferten Seed-Konfiguration. Sie ist die verbindliche Matrix je Rolle.

SystemAdmin – **alle** Berechtigungen (uneingeschränkt).

IT-Administrator (ItAdmin) – **alle Berechtigungen außer** `system.backup.manage` und `identity.permissions.manage`. (Kann also Benutzer/Rollen verwalten, aber nicht das Berechtigungsschema selbst ändern und keine Backups steuern.)

CMDB-Administrator (CmdbAdmin) `cmdb.ci.read`, `cmdb.ci.create`, `cmdb.ci.update`, `cmdb.ci.delete`, `cmdb.ci.lifecycle.manage`, `cmdb.class.manage`, `cmdb.relation.manage`, `cmdb.import`, `cmdb.export`, `cmdb.baseline.manage`, `reporting.read`, `reporting.create`, `licensing.read`, `contracts.read`, `discovery.read`, `dr.read`.

Service-Desk-Agent L1 (SdAgentL1) `tickets.read`, `tickets.create`, `tickets.update`, `tickets.close`, `tickets.escalate`, `ticketing.csat.submit`, `contacts.manage`, `ticketing.service_catalog.read`, `cmdb.ci.read`, `reporting.read`, `knowledge.read.public`, `knowledge.read.internal`, `knowledge.author`.

Service-Desk-Agent L2/L3 (SdAgentL2) – L1 plus: `tickets.assign`, `tickets.delete`, `tickets.export`, `notifications.mailbox.admin`, `ticketing.service_catalog.manage`, `cmdb.ci.update`, `cmdb.relation.manage`, `change.read`, `change.create`, `reporting.create`, `workflow.read`, `workflow.manage`, `knowledge.review`, `knowledge.publish`.

Netzwerk-Administrator (NetworkAdmin) `ipam.prefix.read`, `ipam.prefix.manage`, `ipam.ip.read`, `ipam.ip.assign`, `ipam.ip.free`, `ipam.vrf.manage`, `ipam.dhcp.manage`, `ipam.dhcp.import`, `ipam.scan.trigger`, `ipam.ip.confirm`, `ipam.export`, `cmdb.ci.read`, `cmdb.ci.create`, `reporting.read`, `discovery.read`, `discovery.manage`, `discovery.collector.manage`, `discovery.result.review`.

Change Manager (ChangeManager) `change.read`, `change.create`, `change.approve`, `change.reject`, `change.implement`, `change.emergency`, `cmdb.ci.read`, `tickets.read`, `reporting.read`, `licensing.read`, `contracts.read`, `workflow.read`, `workflow.manage`, `dr.read`, `dr.manage`.

CIO (strategisch, überwiegend lesend) `reporting.read`, `reporting.create`, `reporting.manage`, `cmdb.ci.read`, `tickets.read`, `tickets.read.all`, `ipam.prefix.read`, `ipam.ip.read`, `audit.read`, `licensing.read`, `contracts.read`, `workflow.read`, `dr.read`, `security.report.read`, `security.check.read`, `knowledge.read.public`, `knowledge.read.internal`, `bcm.report.read`, `bcm.plan.read`.

Endbenutzer (Enduser) `tickets.self.read`, `tickets.create`, `ticketing.csat.submit`, `ticketing.service_catalog.read`, `knowledge.read.public`.

Datenschutzbeauftragter (DSB) `audit.read`, `audit.export`, `reporting.read`, `identity.users.read`.

Informationssicherheitsbeauftragter (ISB) – **alle** `security.*`-Rechte plus `cmdb.ci.read`, `reporting.read`.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

BCM-Beauftragter (BCMB) – **alle** `bcm.*`-Rechte plus `cmdb.ci.read`, `cmdb.ci.create`, `cmdb.relation.manage`, `security.check.read`, `dr.read`, `reporting.read`.

5 Besondere Hinweise

- **„Administrator"** im Sinne der Produktlizenz = Benutzer mit `system.config.manage`. Das Administrator-Limit der Lizenz greift beim Zuweisen dieses Rechts (HTTP 402, siehe ID-43).
- **Ticket-Sichtbarkeit:** `tickets.read` ist auf die eigenen **Support-Gruppen** beschränkt; `tickets.read.all` hebt den Gruppen-Filter auf.
- **Konvergenz:** Neue Systemrollen und Berechtigungen werden bei jedem Start additiv nachgezogen; bestehende (Rolle, Recht)-Zuordnungen bleiben unangetastet.
- **Eigene Rollen:** Zusätzliche Rollen sind über `identity.roles.manage` möglich; die Vergabe einzelner Rechte erfolgt über `identity.permissions.manage`.

5.1 Funktionstrennung (Vier-Augen-Prinzip)

Drei Freigaben sind seit **2026.07.14.1845** als feste Regel im System verankert – sie lassen sich auch dann nicht umgehen, wenn eine Person beide Rechte besitzt:

Vorgang	Regel
Änderungsantrag (Change)	Der **Antragsteller** kann seinen eigenen Antrag **nicht** freigeben. Ausnahme: **Notfall-Changes** – dort zählt Geschwindigkeit; die Freigabe wird über die Freigabe-Historie nachvollziehbar dokumentiert (ITIL-Praxis).
Wissensartikel	Öffentliche Artikel müssen von einer **anderen** Person freigegeben werden als dem Autor bzw. Einreicher.
Prozessdefinition	Die Freigabe erfolgt durch eine **andere** Person als den Ersteller; zusätzlich sind Pflege (`process.manage`) und Freigabe (`process.approve`) getrennte Rechte.

Bis zu dieser Version war die Trennung beim Change **nur über die Rollenvergabe** hergestellt: Wer `change.create` **und** `change.approve` besaß, konnte den CAB-Beschluss an sich selbst vorbei durchführen. Das ist nun ausgeschlossen (ISO 27001 A.5.3 „Aufgabentrennung", A.8.32 „Änderungssteuerung").

5.2 Nach dem Update: einmalige Neuanmeldung erforderlich

Die Berechtigungen stehen im Anmelde-Token. Nach dem Update auf **2026.07.14.1845** müssen sich alle Benutzer **einmal neu anmelden**, damit die neuen Rechte (u. a. `ticketing.config.manage`, `process.manage`, `process.approve`) wirksam werden. Ohne Neuanmeldung arbeiten sie weiter mit den Rechten aus ihrem alten Token und erhalten bei den betroffenen Aktionen die Meldung „Keine Berechtigung" (HTTP 403).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6 Bedienung in der Anwendung

6.1 Berechtigungsmatrix (Rollen × Berechtigungen)

> 💻 **Nur im Client:** *Fußbereich ▸ Einstellungen ▸ Rollen & Berechtigungen* · 🔑 `identity.roles.read` zum Ansehen, `identity.permissions.manage` zum Ändern — Das Portal weist Rollen unter *Verwaltung ▸ Benutzer und Rollen* zu, stellt sie aber nicht zusammen.

🔒 **Pflichtreihenfolge — Systemrollen sind schreibgeschützt.** Eine eigene Rolle muss deshalb existieren, bevor Sie Rechte setzen können; die Kontrollkästchen einer Systemrolle bleiben deaktiviert, und ein Hinweisbanner sagt das auch.

1. Klicken Sie auf *Neue Rolle* und vergeben Sie einen Namen. Die neue Rolle ist sofort ausgewählt und bearbeitbar.
2. Setzen Sie die benötigten Rechte per Kontrollkästchen – die Liste ist **nach Modul gruppiert**.
3. Klicken Sie auf *Speichern*.
4. Weisen Sie die Rolle anschließend den Benutzern zu (ID-41, Abschnitt 2).

✅ **Ergebnis:** Die Rolle steht zur Auswahl. Beachten Sie: Betroffene Benutzer müssen sich **neu anmelden**, damit die Rechte im Token ankommen (Abschnitt 5.2).

Eine bestehende Rolle ansehen: Wählen Sie sie oben aus; die Berechtigungen erscheinen darunter.

Eine eigene Rolle löschen: Klicken Sie auf *Rolle löschen* – nur bei eigenen Rollen und mit Rückfrage.

> **Tipp:** Fangen Sie nicht bei null an. Sehen Sie sich die Systemrolle an, die Ihrem Zweck am nächsten kommt (Abschnitt 4), und bilden Sie deren Rechte in Ihrer eigenen Rolle nach – abzüglich dessen, was die Person wirklich nicht braucht.

6.2 Mandanten-Verwaltung (nur SuperAdmin / Plattform-Administrator)

> 💻 **Client:** *Fußbereich ▸ Einstellungen ▸ Mandanten* · 🌐 **Portal:** *Verwaltung ▸ Mandanten und Module* · 🔑 `tenant.manage`

🔒 **Pflichtreihenfolge — Löschen ist nur nach vorheriger Archivierung möglich.** Das ist keine Schikane, sondern die einzige Stelle, an der ein Irrtum noch auffällt, bevor die Datenbank weg ist.

- **Mandant anlegen:** Kürzel, Name und optional eine Admin-E-Mail angeben. Beim Anlegen wird eine **eigene Datenbank** für den Mandanten bereitgestellt. Voraussetzung dafür ist, dass die Datenbank-Rolle der Anwendung Datenbanken anlegen darf – siehe Betriebshandbuch (ID-50), Abschnitt zur Mandanten-Provisionierung. Fehlt das Recht, meldet das Anlegen einen Fehler („keine Berechtigung, um Datenbank zu erzeugen“).
- **Archivieren** blendet den Mandanten aus dem Betrieb aus (Daten bleiben erhalten).
- **Löschen** entfernt den Mandanten endgültig (Datenbank wird verworfen) und ist nur **nach vorheriger Archivierung** und mit Bestätigung möglich.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Lizenz-Grenze: Beim Anlegen prüft Ordivis Platform die aktive Produktlizenz. Ist die in der Lizenz festgelegte **Höchstzahl an Mandanten** erreicht, wird das Anlegen abgelehnt und ein Hinweis eingeblendet („Die aktive Lizenz erlaubt maximal *X* Mandanten … für eine Erweiterung den Vertrieb kontaktieren"). Eine Lizenz ohne Mandanten-Begrenzung (bzw. die Testphase) hat kein Limit.

6.3 Mandanten-Umschalter

Gehört ein Benutzer zu mehreren Mandanten – der **SuperAdmin** und Plattform-Administratoren erreichen **alle** aktiven Mandanten –, erscheint in der Titelleiste (Gebäude-Symbol) ein **Umschalter**. Ein Klick öffnet die Mandantenliste; die Auswahl wechselt **ohne Neustart** in den Zielmandanten (die Navigation passt sich den dort geltenden Rechten an).

6.4 DSGVO-Betroffenenrechte

Der Datenschutzbeauftragte (Recht `privacy.manage`) kann je Mandant die Betroffenenrechte bedienen: **Datenauskunft** (strukturierter Export aller personenbezogenen Daten einer Person), **Löschung** (Anonymisierung unter Wahrung der Nachweispflichten) sowie das **Verarbeitungsverzeichnis** (Art. 30 DSGVO).

Verwandte Themen

- **ID-41 – Modul Identität, Benutzer & Rollen** – Rollen anlegen und zuweisen
- **ID-42 – Modul Mandantenfähigkeit** – Wirkung der Mandantengrenze auf Rechte
- **ID-27 – Modul Reporting & Dashboards** – warum Auswertungen je Rolle abweichen
- **ID-90 – Anhang – Glossar & Referenzen** – Kurzreferenz der wichtigsten Rechte

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL VI · BETRIEB

Betriebshandbuch – Installation & Serverbetrieb

Installation, Update, Sicherung, Härtung und Fehlerbehebung

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-50 Betrieb

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Betriebshandbuch – Installation & Serverbetrieb

Dieses Dokument richtet sich an **IT-Betrieb / Server-Administratoren**. Es beschreibt Installation, Aktualisierung, Sicherung, Härtung und Fehlerbehebung. Architektur und Voraussetzungen stehen in ID-02.

INHALT

1 Deployment-Grundsätze
2 Installation (Reihenfolge)
2.1 Server vollständig bereinigen (für eine saubere Neuinstallation)
2.2 Installations-Assistent (Bildschirm für Bildschirm)
2.2.1 Die drei Installationspakete
2.2.2 Collector-Installation (Standort-Rechner)
2.3 PostgreSQL-Tuning bei der Installation
3 Konfiguration (appsettings)
3.1 Signaturschlüssel der Anmelde-Token (Pflicht)
3.2 Betrieb hinter einem Reverse-Proxy (Pflichtangabe seit 2026.08)
3.3 Wetter und amtliche Unwetterwarnungen einrichten
3.4 Windows-Anmeldung im Web-Portal (seit 2026.08)
3.5 Sitzungsdauer (seit 2026.08)
4 Datenbank-Migrationen
4.1 Mandanten-Provisionierung (eigene Datenbank je Mandant)
5 In-Place-Update & Versionierung
5.1 Delta-Update (signiert, client-gesteuert)
Mehrere Installationen aus einem Client
5.2 Update-Center & Auto-Update-Policy
6 Sicherung & Wiederherstellung
6.1 Verschlüsselte Sicherungen (7z, AES-256)
6.2 Weitere Sicherung & Wiederherstellung
6.3 Anhänge liegen in der Datenbank
7 Sicherheit & Härtung
7.1 HTTPS einrichten
7.2 Client-Verbindung (API-Adresse)
7.3 TLS-Version
7.4 Zugänge für Geräte und Programme (Token)
8 Skripte & Encoding
9 Fehlerbehebung (Auszug)
10 Logs
10.1 Einstellen im Client (empfohlener Weg)
10.2 Einstellen in der Datei (appsettings.json)
10.2a Format Json (CLEF) für SIEM und Log-Agenten
10.3 Zentrale Sammlung (mehrere Server, SIEM)
10.4 Weitere Protokolle
10.5 Support-Paket – ein Klick statt zehn Rückfragen
Verwandte Themen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1 Deployment-Grundsätze

- **Kein Docker.** Betrieb nativ auf Windows Server.
- Komponenten als **Windows-Dienste**: API und Self-Service-Portal (:8080). Die Netzscans laufen **in der API**; ein eigener Dienst dafür existiert nur im Sonderfall eines entfernten Netzes (Discovery-Collector).
- **PostgreSQL** nativ; **WinUI-Client** unpackaged je Arbeitsplatz.

2 Installation (Reihenfolge)

1. **PostgreSQL** installieren.

   > **Voraussetzung:** Installer mit `--install_runtimes 1` aufrufen, sonst fehlt die VC++-Runtime.

2. **Datenbank & Rolle** anlegen (`infradesk_app`), Rechte vergeben.
3. **API-Dienst** installieren (Setup/Installer), `appsettings` konfigurieren (Connection-String, `CredentialVault:EncryptionKey` als Secret/Umgebungsvariable).
4. **Portal** als Dienst einrichten.
5. **Migrationen** ausführen (siehe Abschnitt 4).
6. **Erststart/Onboarding** durchführen (ID-40).

> **Kein Discovery-Dienst auf dem Server.** Die serverseitigen Scans laufen im API-Dienst selbst; es gibt dafür keinen eigenen Windows-Dienst. Ein separater Dienst existiert nur für den **Collector**, und zwar am entfernten Standort – nicht in der Zentrale (Abschnitt 2.2.2).

> **Wichtig:** Dienste mit `New-Service` anlegen, nicht mit `sc.exe create` – letzteres scheitert unter PowerShell 5.1 still am Pfad-Quoting.

2.1 Server vollständig bereinigen (für eine saubere Neuinstallation)

> **Wichtig – die normale Deinstallation genügt dafür nicht.** Sie entfernt nur die Dienste und Programmdateien; **PostgreSQL und die Ordivis-Datenbank bleiben erhalten** (gewollt, damit ein Update keine Daten verliert). Eine spätere Neuinstallation **verwendet diese Datenbank weiter**: Enthält sie bereits einen Mandanten, gilt das System als eingerichtet – der **Erststart-Assistent erscheint dann nicht**, und die bisherigen Anmeldedaten gelten weiter. Der Installer weist am Ende darauf hin, wenn er eine vorhandene Datenbank übernommen hat. Wer wirklich „von Null" beginnen will, nutzt die folgende Bereinigung.

Für eine echte Neuinstallation „von Null" – oder um Reste eines fehlgeschlagenen Versuchs zu entfernen – gibt es das Skript `cleanup-server.ps1` (im Setup-Paket unter `deploy\inno\`). Es entfernt **vollständig**:

- die Windows-Dienste `InfraDesk.API` , `InfraDesk.Portal` und die PostgreSQL-Dienste,
- führt den **PostgreSQL-Uninstaller** aus,
- löscht die Installations- und Datenverzeichnisse (`…\Ordivis Platform` , `…\PostgreSQL` , `C:\ProgramData\InfraDesk`) inklusive eines individuell gewählten Backup-Verzeichnisses,
- entfernt Firewall-Regeln, Machine-Umgebungsvariablen und die geplante Aufgabe `InfraDesk-DailyBackup` .

Die .NET-Runtime bleibt bewusst erhalten.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

> ⚠️ **Unwiderruflich:** Die Ordivis-Datenbank **und alle Sicherungen** werden gelöscht. Vorher bei Bedarf eine Kopie außerhalb der genannten Verzeichnisse ablegen.

Als **Administrator** ausführen (fragt einmal „ja/nein"):

```
powershell -ExecutionPolicy Bypass -File .\cleanup-server.ps1
```

Ohne Rückfrage (z. B. für Automatisierung): zusätzlich `-Force`. Der **Installer erkennt Altbestände auch selbst** und bietet die Bereinigung vor der Installation an – der manuelle Aufruf ist also optional, für einen echten Neustart aber die saubere Variante.

2.2 Installations-Assistent (Bildschirm für Bildschirm)

Der Windows-Installer (`Ordivis-Setup-<Version>.exe` , als **Administrator** ausführen) führt in wenigen Schritten durch die Einrichtung. Alle Passwörter werden zufällig vorbelegt und am Ende in `C:\ProgramData\InfraDesk\setup-credentials.txt` gesichert.

Die Abbildungen 1 bis 3 zeigen die ersten Seiten des Assistenten.

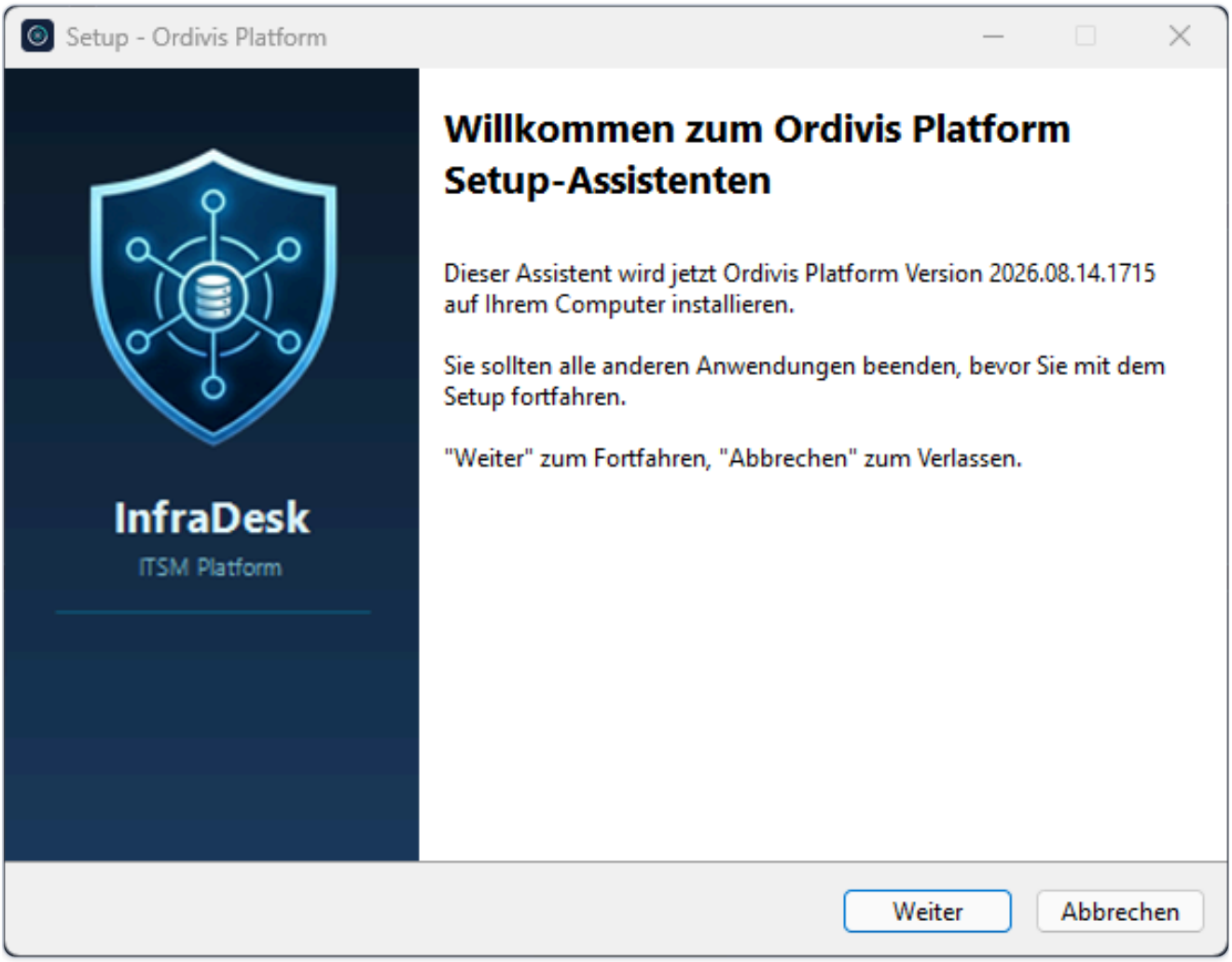

Abb. 1 – Startseite des Assistenten. Es folgen Lizenzvereinbarung und die Auswahl des Installationstyps.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

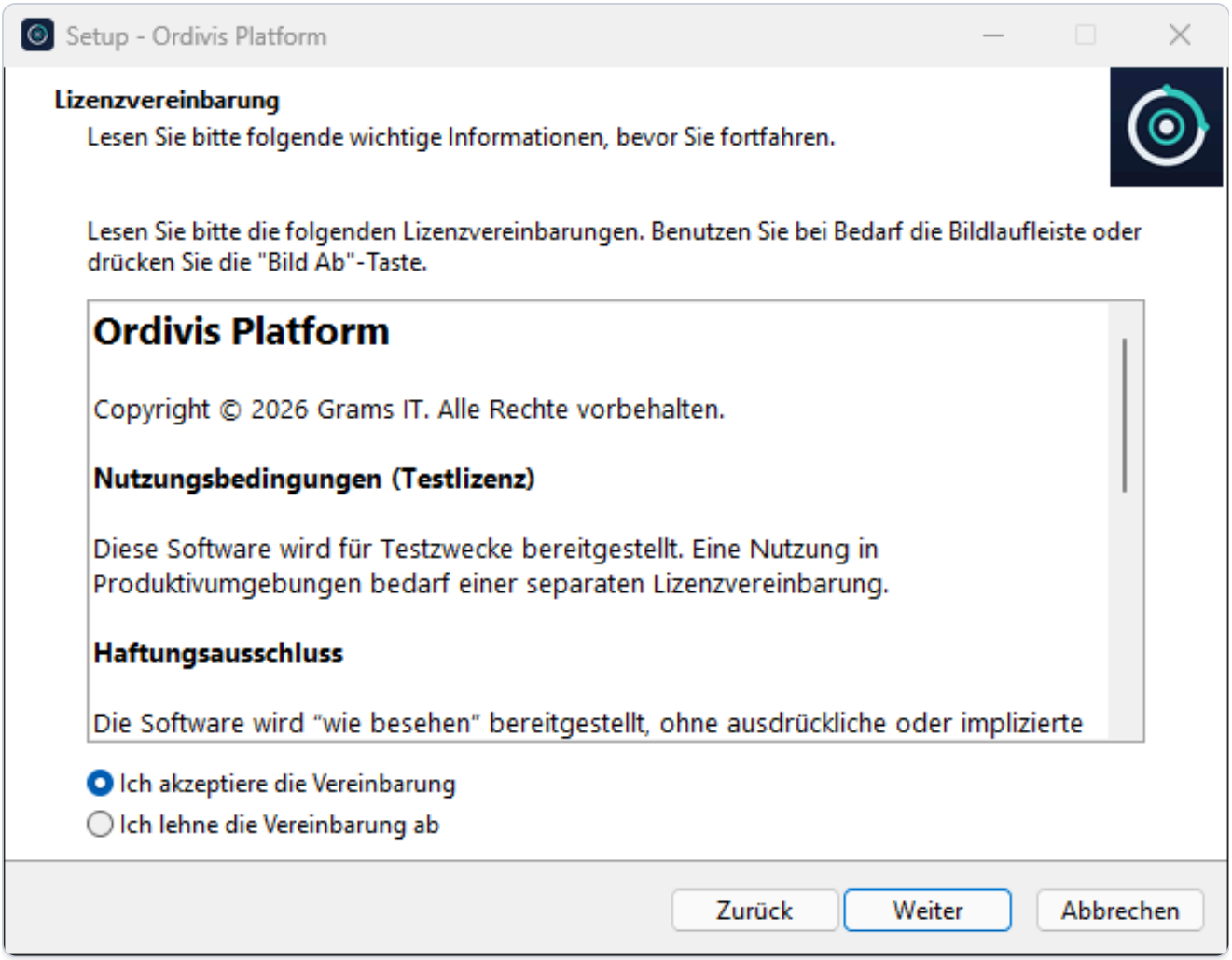

Abb. 2 – **Lizenzvereinbarung**: Zum Fortfahren „Ich akzeptiere die Vereinbarung" wählen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

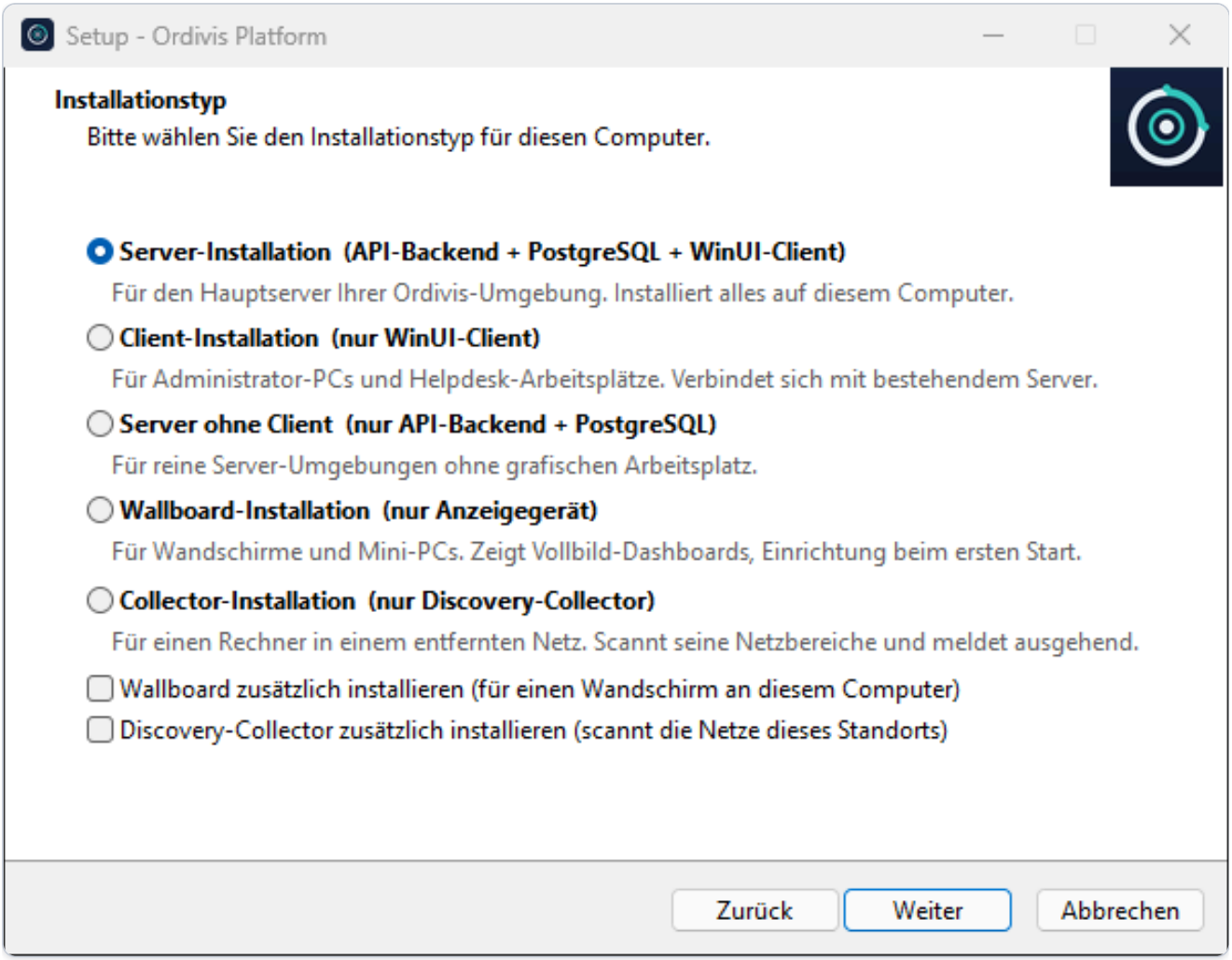

Abb. 3 – **Installationstyp**: *Server* (API + PostgreSQL + Client), reine *Client*-Installation, *Server ohne Client* oder *Wallboard-Installation*. (Die Abbildung zeigt noch den Stand vor 2026.07.28.1912 mit drei Typen.)

Seit Version 2026.07.28.1912 gibt es einen **vierten Installationstyp: Wallboard-Installation**. Er ist für Geräte gedacht, die *nur* als Anzeigebildschirm dienen – ein Rechner am Wandschirm im Teamraum, ohne Arbeitsplatz-Client und ohne Server. Installiert wird dann ausschließlich die Wallboard-Anwendung samt der benötigten Windows-App-Laufzeit.

Soll ein Rechner **beides** sein – Arbeitsplatz oder Server *und* zusätzlich einen Wandschirm bedienen –, setzen Sie bei den Installationstypen *Server* oder *Client* einfach den **Zusatzhaken für das Wallboard**.

> **Hinweis zu älteren Setup-Dateien.** In Setup-Dateien vor 2026.07.28.1912 war das Wallboard zwar enthalten, ließ sich aber nicht auswählen – der Assistent übersprang die Komponentenauswahl und setzte hart nur die drei bekannten Installationstypen. Wer ein Anzeigegerät einrichten will, braucht daher mindestens diese Version. Ebenfalls behoben: Bei einer reinen Wallboard-Installation wird die Windows-App-Laufzeit jetzt mit eingerichtet; ohne sie wäre die Anwendung nach der Installation nicht gestartet.

Seit **2026.07.29** gibt es einen **fünften Installationstyp: Collector-Installation** – der Discovery-Collector für einen Rechner in einem entfernten Netz (siehe Kapitel **Modul Discovery**, Abschnitt 6). Auch er ist als **Zusatzhaken** neben *Server* oder *Client* wählbar, wenn Zentrale und Collector auf derselben Maschine laufen sollen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Wallboard und Collector sind damit **Zusatzrollen, keine Installationsarten**: Beide lassen sich einzeln hinzunehmen und einzeln weglassen. Wer nur eines von beiden braucht, wählt genau das.

2.2.1 Die drei Installationspakete

Nicht jede Rolle braucht das vollständige Paket:

Paket	Größe	Enthält	Für
`Ordivis-Setup-<Version>.exe`	~630 MB	Server, PostgreSQL, Portal, Client; Wallboard und Collector zuwählbar	die Zentrale
`Ordivis-Wallboard-Setup-<Version>.exe`	~90 MB	nur die Wallboard-Anwendung	Anzeigegeräte
`Ordivis-Collector-Setup-<Version>.exe`	~70 MB	nur den Collector-Dienst	Standort-Rechner in fremden Netzen

Die beiden schlanken Pakete gibt es aus einem handfesten Grund: Einem Kunden oder einer Außenstelle das 630-MB-Paket zu schicken, hieße dort Server, Datenbank und Portal mitzuliefern – Software, die auf einem Anzeigegerät oder einem Scan-Rechner nichts zu suchen hat.

> **Sie lassen sich nebeneinander installieren.** Jedes Paket trägt eine eigene Kennung. Ein schlankes Paket behandelt eine vorhandene Vollinstallation deshalb **nicht** als Upgrade und nimmt sie beim Deinstallieren **nicht** mit.

2.2.2 Collector-Installation (Standort-Rechner)

Der Collector-Installer verlangt **Administratorrechte** – anders als der Wallboard-Installer, der auch ins Benutzerprofil installieren kann. Grund: Der Collector wird als Windows-Dienst registriert und verschlüsselt sein Scan-Kennwort mit dem Maschinenschlüssel; beides geht nicht ohne Erhöhung.

Er fragt zwei Seiten ab:

1. **Verbindung zur Zentrale** – Adresse der Ordivis-API und der **Anmeldeschein** aus der Collector-Verwaltung. Die Zertifikatsprüfung lässt sich abschalten (nur für selbstsignierte Zertifikate) – als ausdrücklicher Haken, nicht als Textfeld: Über diese Verbindung reist ein Zugang.
2. **Anmeldedaten für den Tiefen-Scan** – ein **lesendes** Windows-Konto, optional. Ohne Konto bleibt es beim Netz-Durchlauf.

Nach der Installation läuft der Dienst **InfraDesk.Discovery.Collector** (verzögerter Automatikstart). Was er tut, steht in `C:\ProgramData\InfraDesk\Logs\worker-*.log`; seine Einstellungen in `<Installationsverzeichnis>\collector\appsettings.json`.

> **Der Dienst startet nur, wenn er etwas tun kann.** Ohne Serveradresse wird er angelegt, aber nicht gestartet – ein Dienst, der beim Start sofort „nicht konfiguriert" meldet, sieht im Ereignisprotokoll wie ein Fehler aus und ist doch nur unfertig eingerichtet.

Die Abbildungen 4 bis 12 zeigen die weiteren Seiten in der Reihenfolge, in der sie erscheinen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

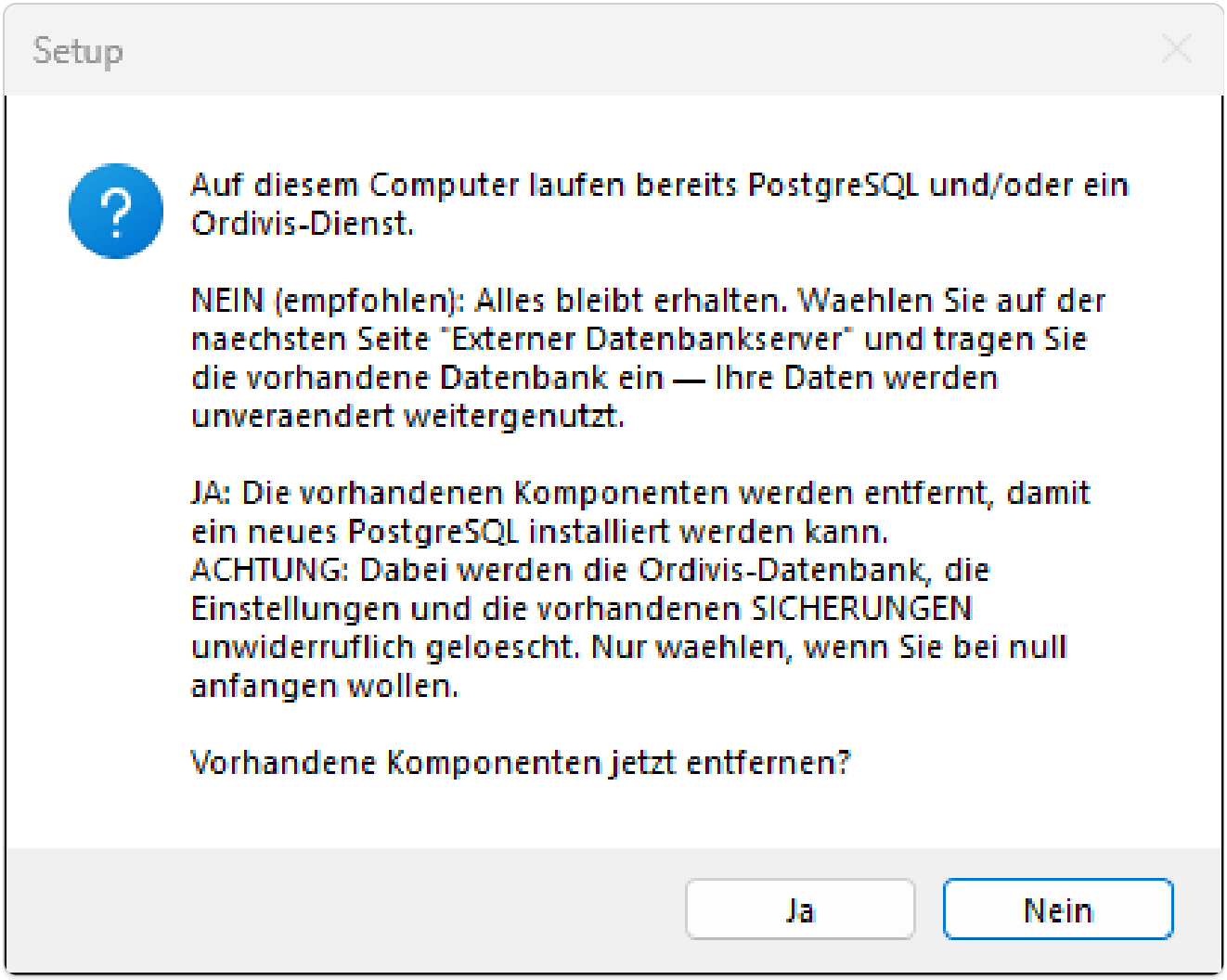

Abb. 4 – Findet das Setup Reste einer früheren Installation, bietet es die **automatische Bereinigung** an (siehe 2.1). Bei einer Erstinstallation erscheint dieser Dialog nicht.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

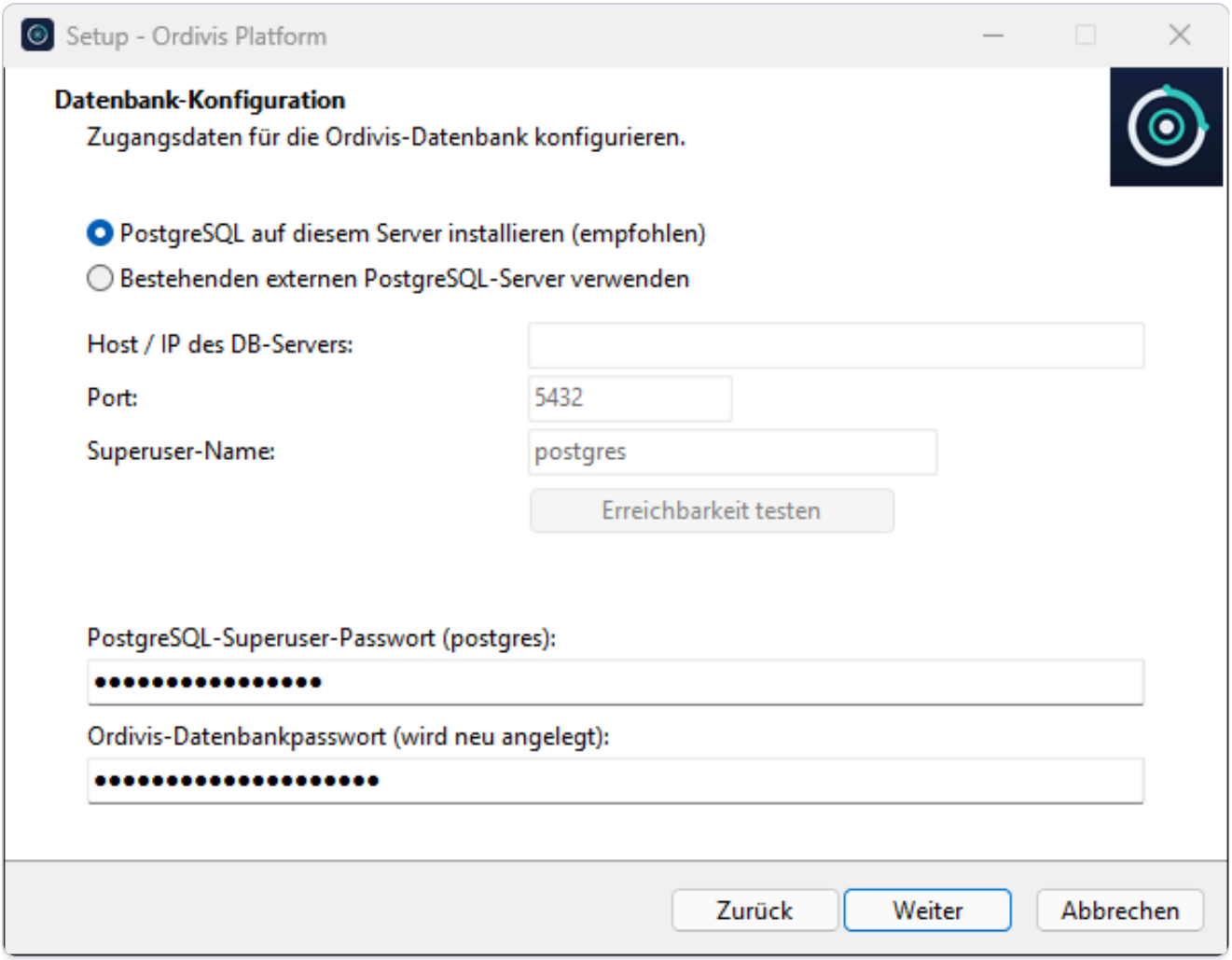

Abb. 5 – **Datenbank**: PostgreSQL lokal installieren (empfohlen) oder einen bestehenden externen Server verwenden. Die Passwörter sind vorbelegt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

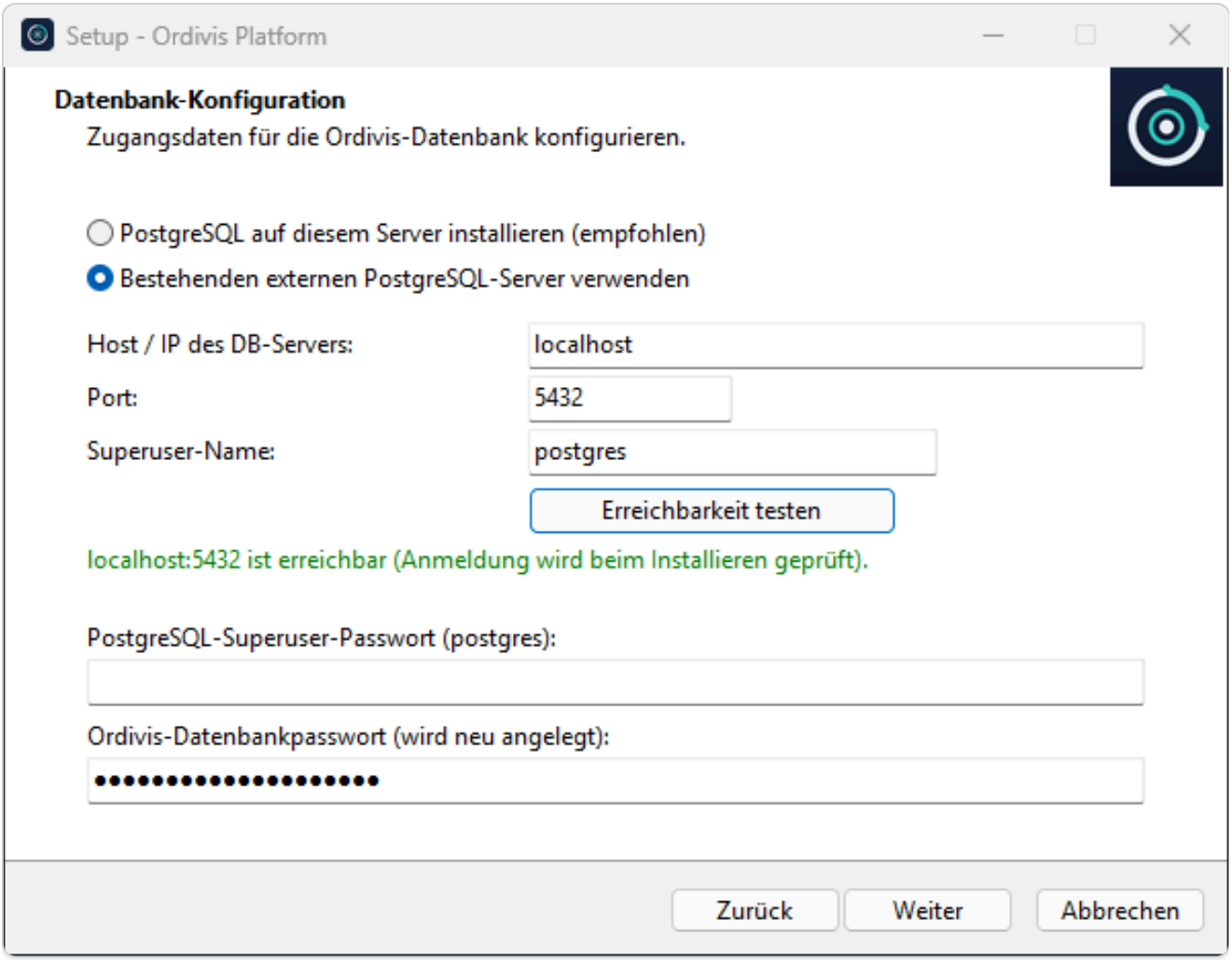

Abb. 6 – Dieselbe Seite mit **bestehendem externem Server**: Host, Port und Superuser-Name werden eingabefähig. **Erreichbarkeit testen** prüft nur, ob der Server antwortet – die Anmeldung selbst wird erst beim Installieren geprüft, und genau das sagt die Meldung auch.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

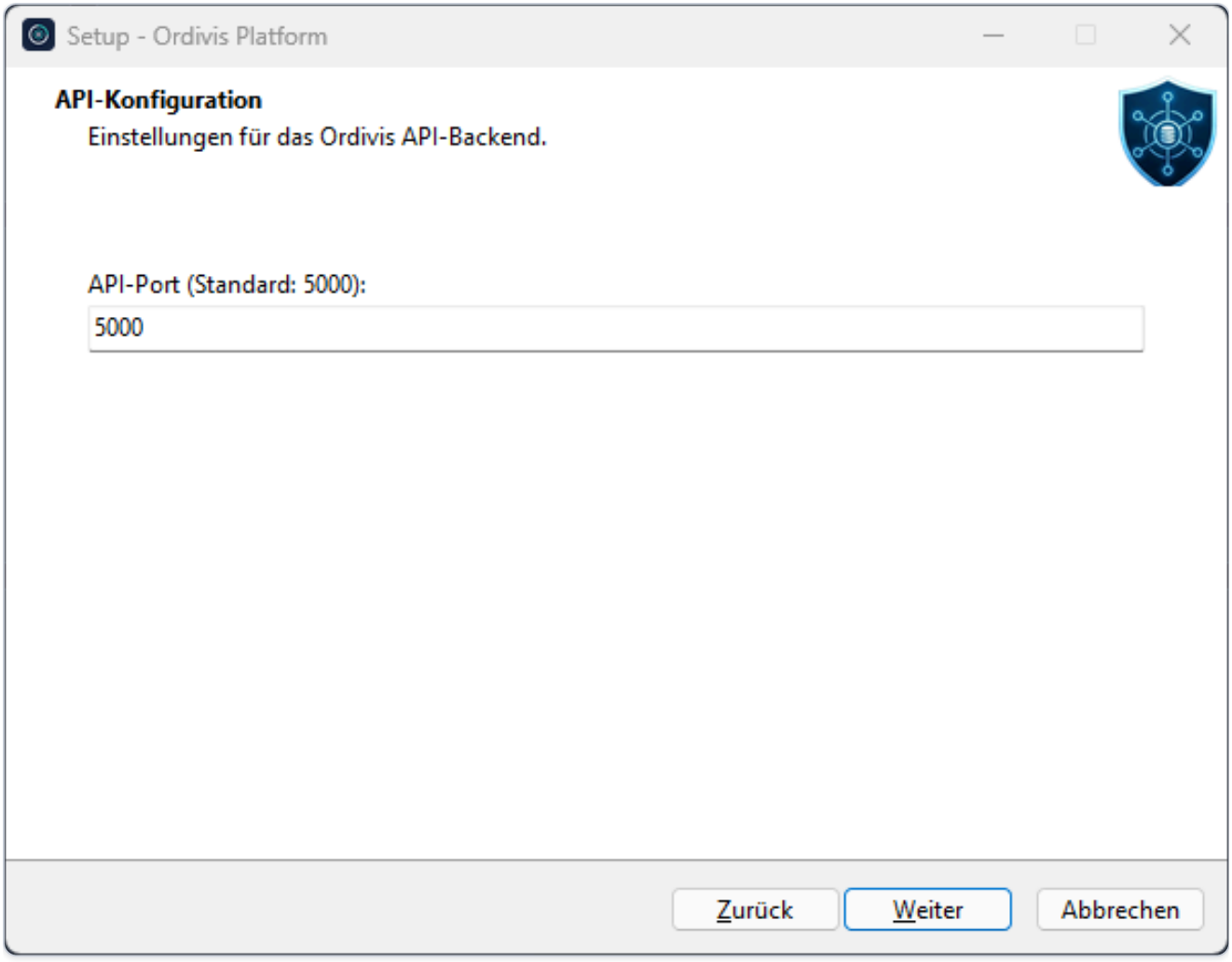

Abb. 7 – **API-Konfiguration**: Port des API-Backends (Standard `5000`). In der Regel unverändert lassen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

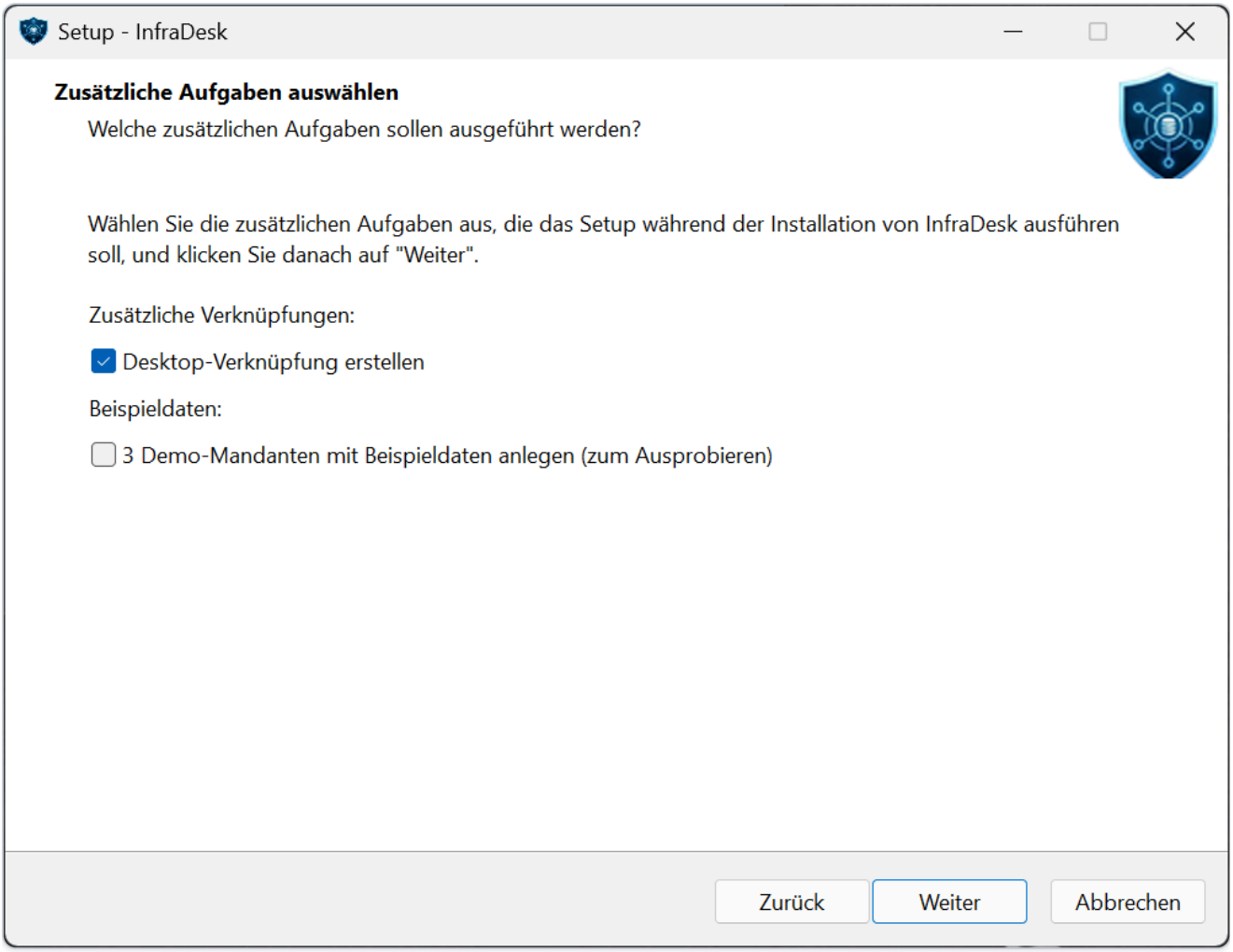

Abb. 8 – **Zusätzliche Aufgaben**: Desktop-Verknüpfung sowie die Option **„3 Demo-Mandanten mit Beispieldaten anlegen"**. Sie füllt drei Vorführ-Mandanten (kleines Unternehmen, Konzern, Kommunalverwaltung) mit anonymen Beispieldaten zum Ausprobieren. **Für Produktivinstallationen deaktiviert lassen.**

Erststart trotz Demo-Mandanten: Die Demo-Mandanten zählen **nicht** als eingerichtetes System. Der Client startet daher auch bei installierten Demo-Mandanten den **Erststart-Assistenten** für den ersten eigenen Mandanten samt **SuperAdmin** (Schritt 6, Erststart/Onboarding). So entsteht immer ein produktiver Mandant mit einem echten SuperAdmin-Konto – unabhängig von den Vorführdaten. Wer die Demo-Mandanten nicht benötigt, lässt die Option ab.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

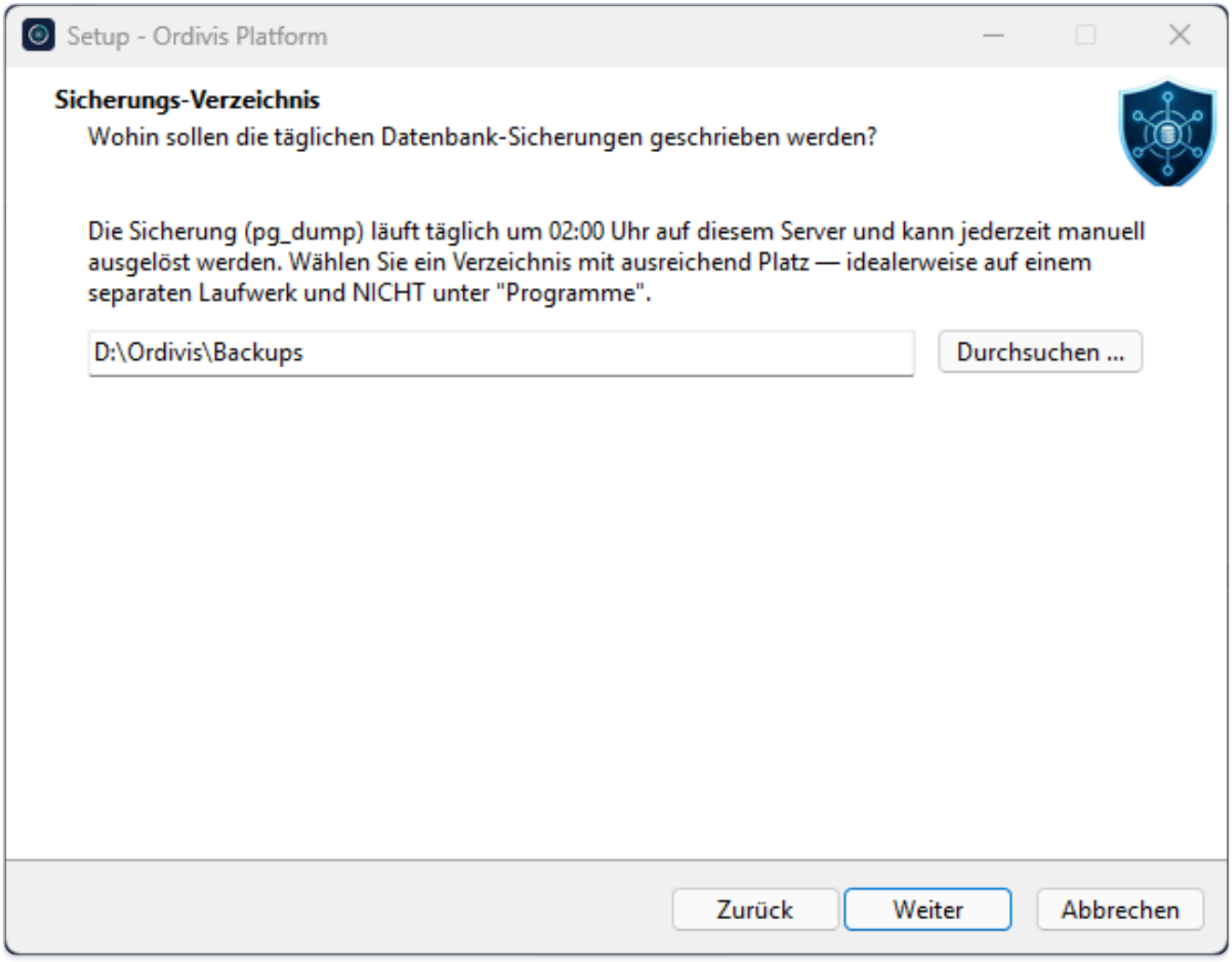

Abb. 9 – **Sicherungs-Verzeichnis** für die tägliche Datenbank-Sicherung (Standard `D:\Ordivis Platform\Backups`, nicht unter „Programme"). Siehe Abschnitt 6 und 4.1.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

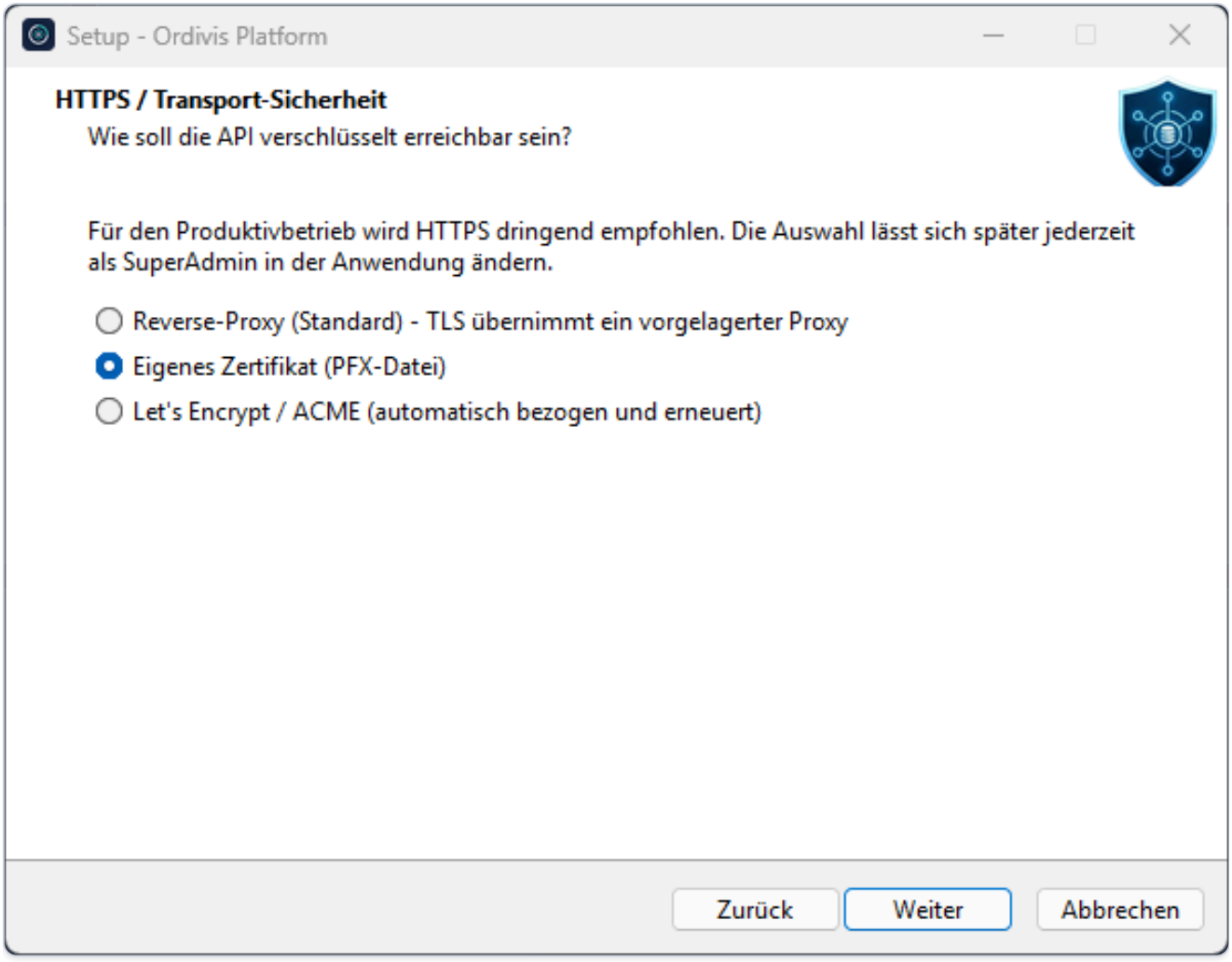

Abb. 10 – **HTTPS / Transport-Sicherheit**: Reverse-Proxy (Vorgabe), eigenes Zertifikat als PFX-Datei oder Let's Encrypt. Die Auswahl lässt sich später jederzeit als SuperAdmin ändern – siehe Abschnitt 7.1.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

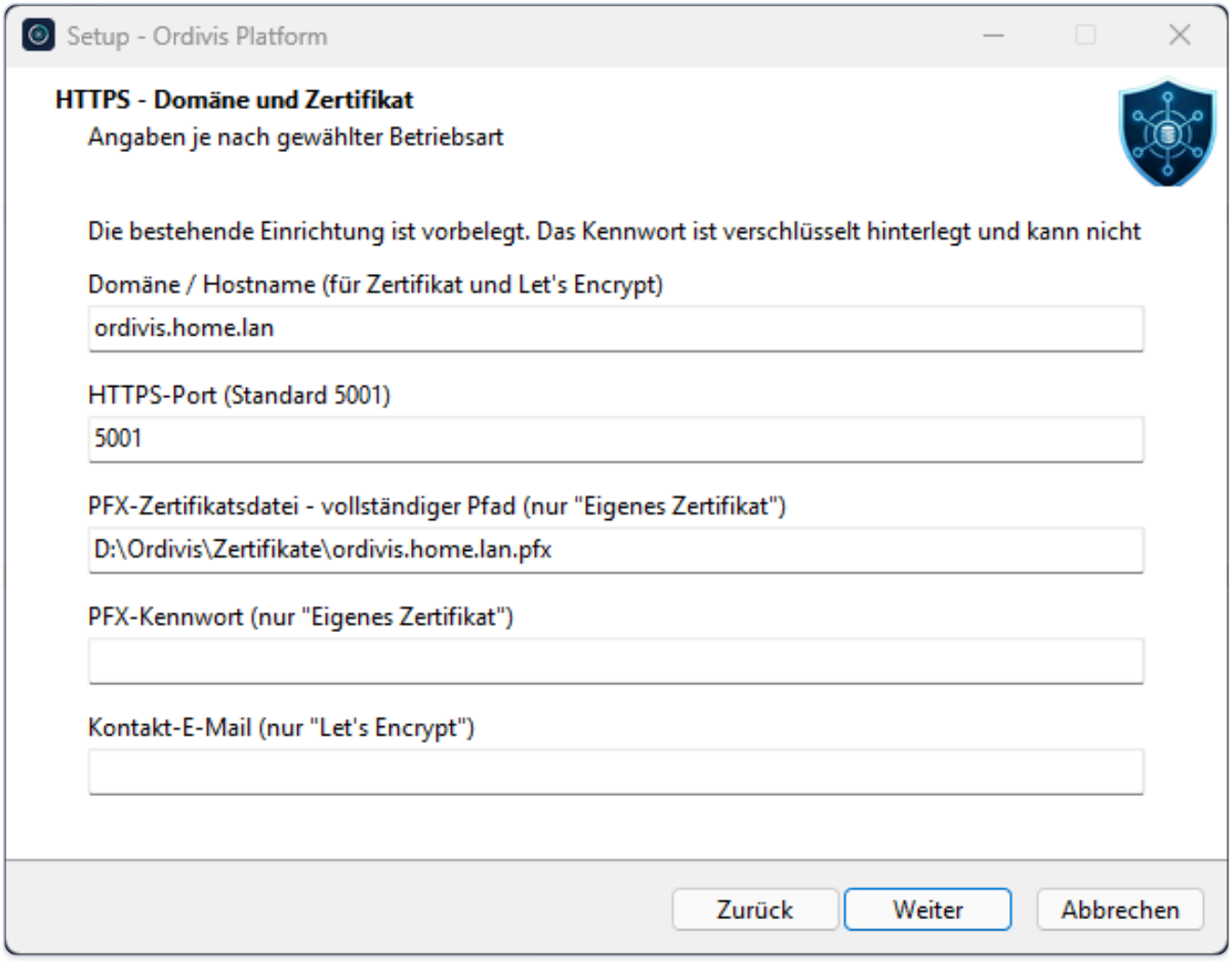

Abb. 11 – **HTTPS – Domäne und Zertifikat**. Auszufüllen sind nur die Felder der gewählten Betriebsart; bei *Reverse-Proxy* wird diese Seite übersprungen. Eine bestehende Einrichtung ist vorbelegt, sodass unverändertes Durchklicken sie unverändert lässt.

Das PFX-Kennwort einer bestehenden Einrichtung wird nicht angezeigt – anders als Domäne, Port und Zertifikatspfad, die vorbelegt erscheinen. Es liegt **DPAPI-geschützt** in der Datei und ist für den Installer, der unter einem anderen Konto läuft, nicht lesbar. **Ein leeres Feld heißt hier „unverändert lassen"**: Der gespeicherte Wert wird übernommen. Nur wer etwas einträgt, ersetzt ihn. Der Assistent sagt das an dieser Stelle auch selbst.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

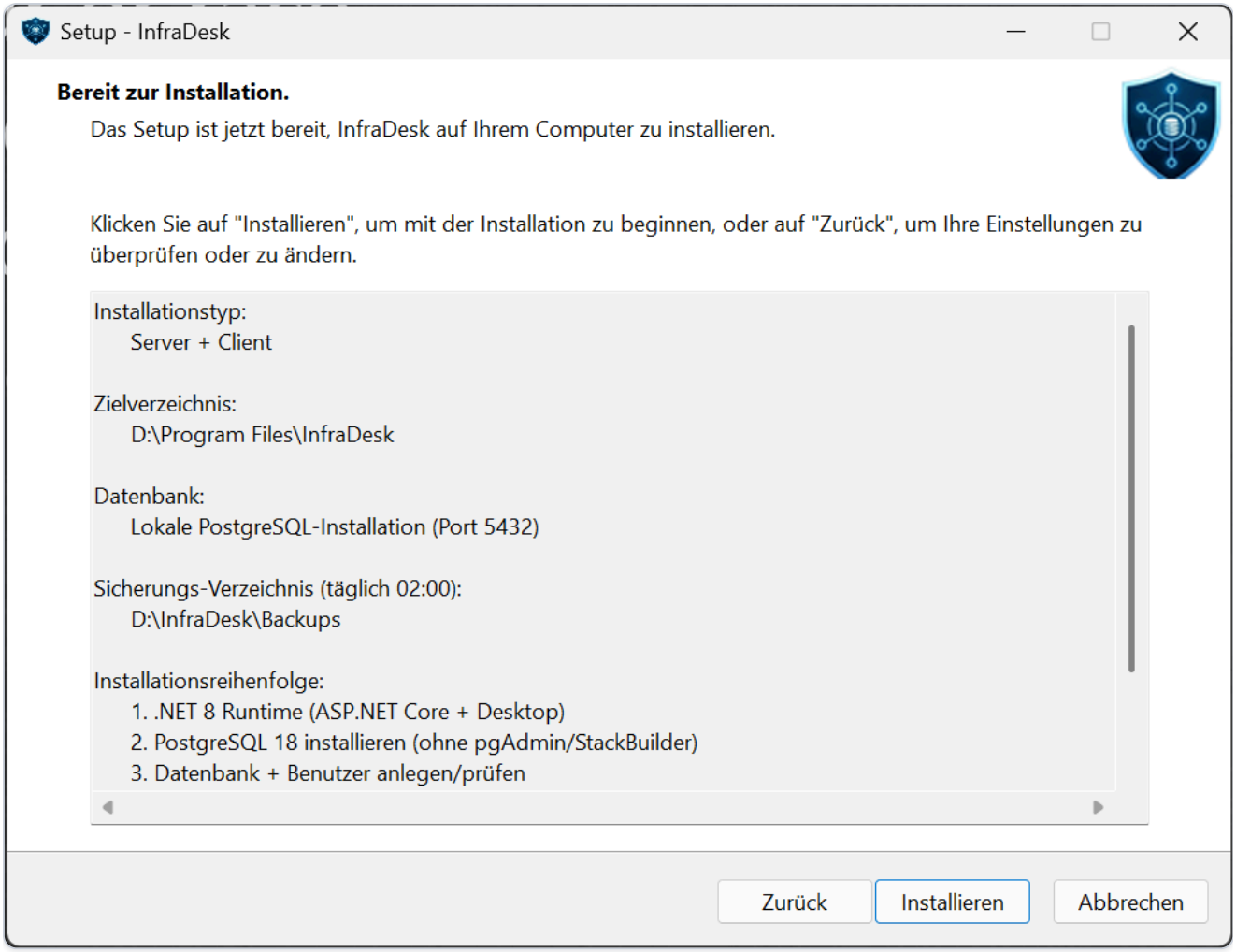

Abb. 12 – **Bereit zur Installation**: Zusammenfassung mit Zielverzeichnis, Datenbank, Sicherungs-Verzeichnis und Installationsreihenfolge.

Nach dem Klick auf **Installieren** läuft die Einrichtung in fünf sichtbaren Schritten ab: (1) .NET-Runtime, (2) PostgreSQL-Datenbankserver, (3) Datenbank und Benutzer anlegen, (4) Windows-Dienste registrieren, (5) Dienst starten und erste Datenbank-Migration. Die beiden längsten Schritte melden sich mit einem ausdrücklichen Hinweis.

Die Abbildungen 13 bis 16 zeigen den Fortschritt und den Abschluss.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

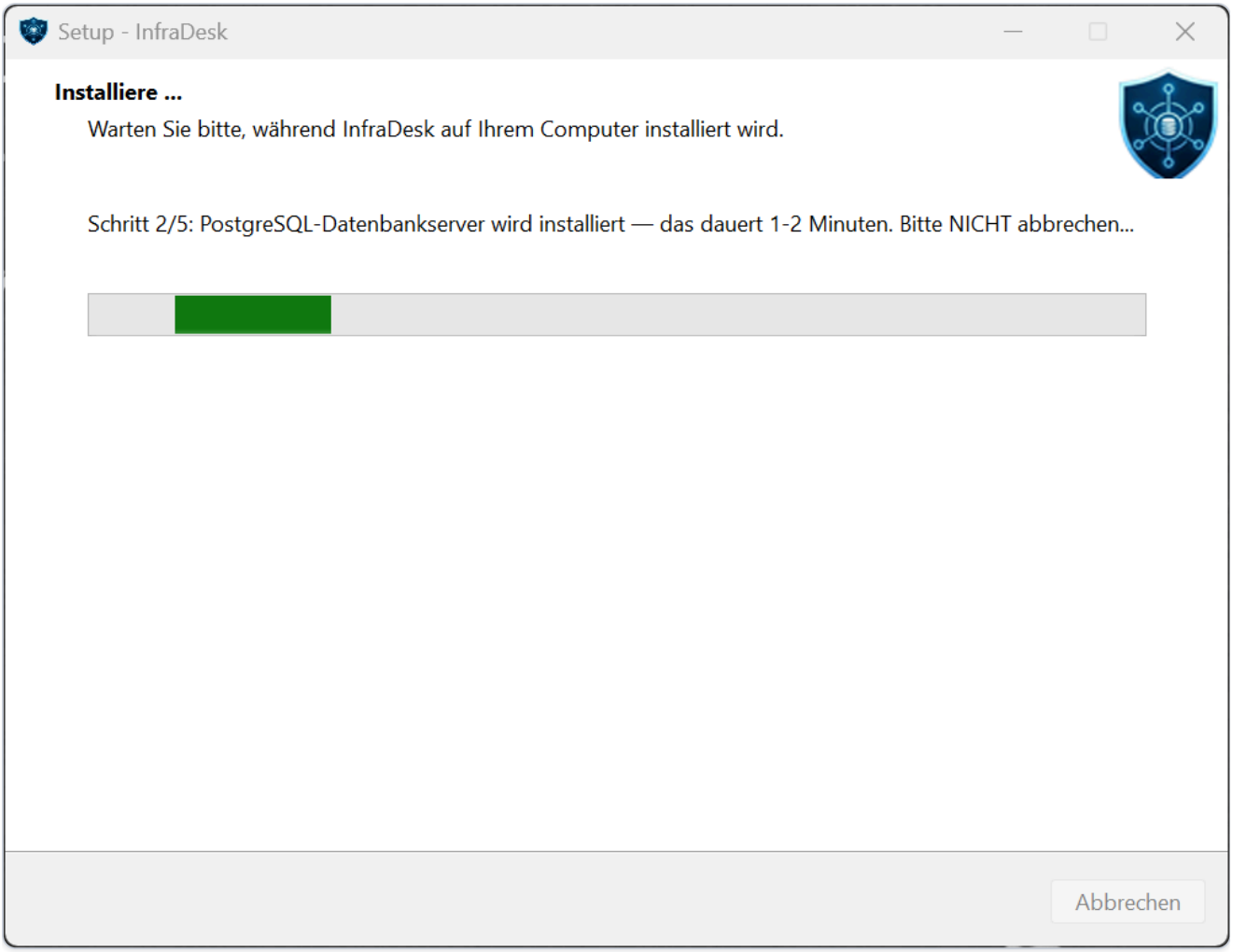

Abb. 13 – **Schritt 2/5**: Der PostgreSQL-Datenbankserver wird installiert (1–2 Minuten). **Bitte nicht abbrechen** – dieser Schritt läuft ohne sichtbares PostgreSQL-Fenster im Hintergrund.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

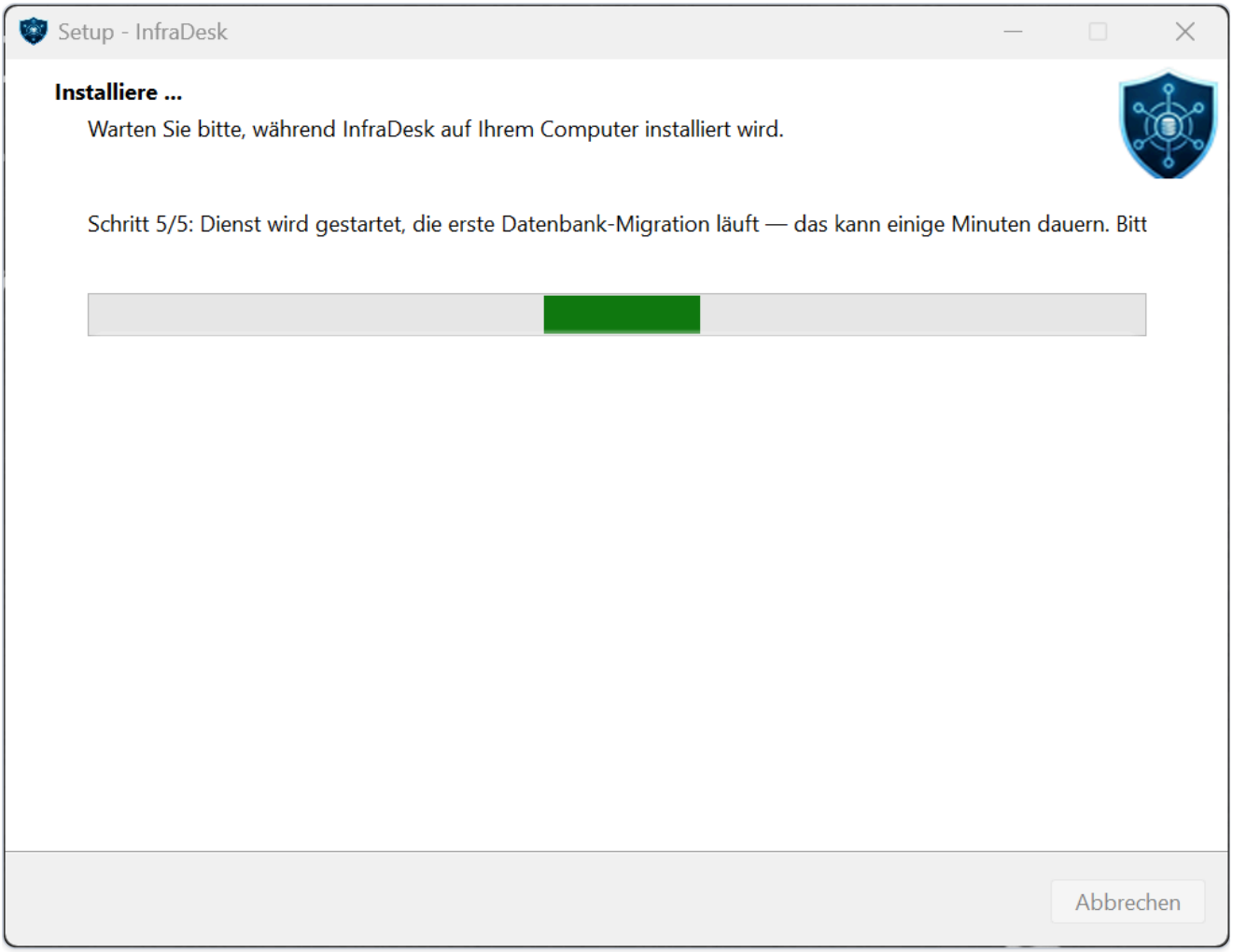

Abb. 14 – **Schritt 5/5**: Die Windows-Dienste starten und die **erste Datenbank-Migration** läuft – das kann einige Minuten dauern. Bitte nicht abbrechen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

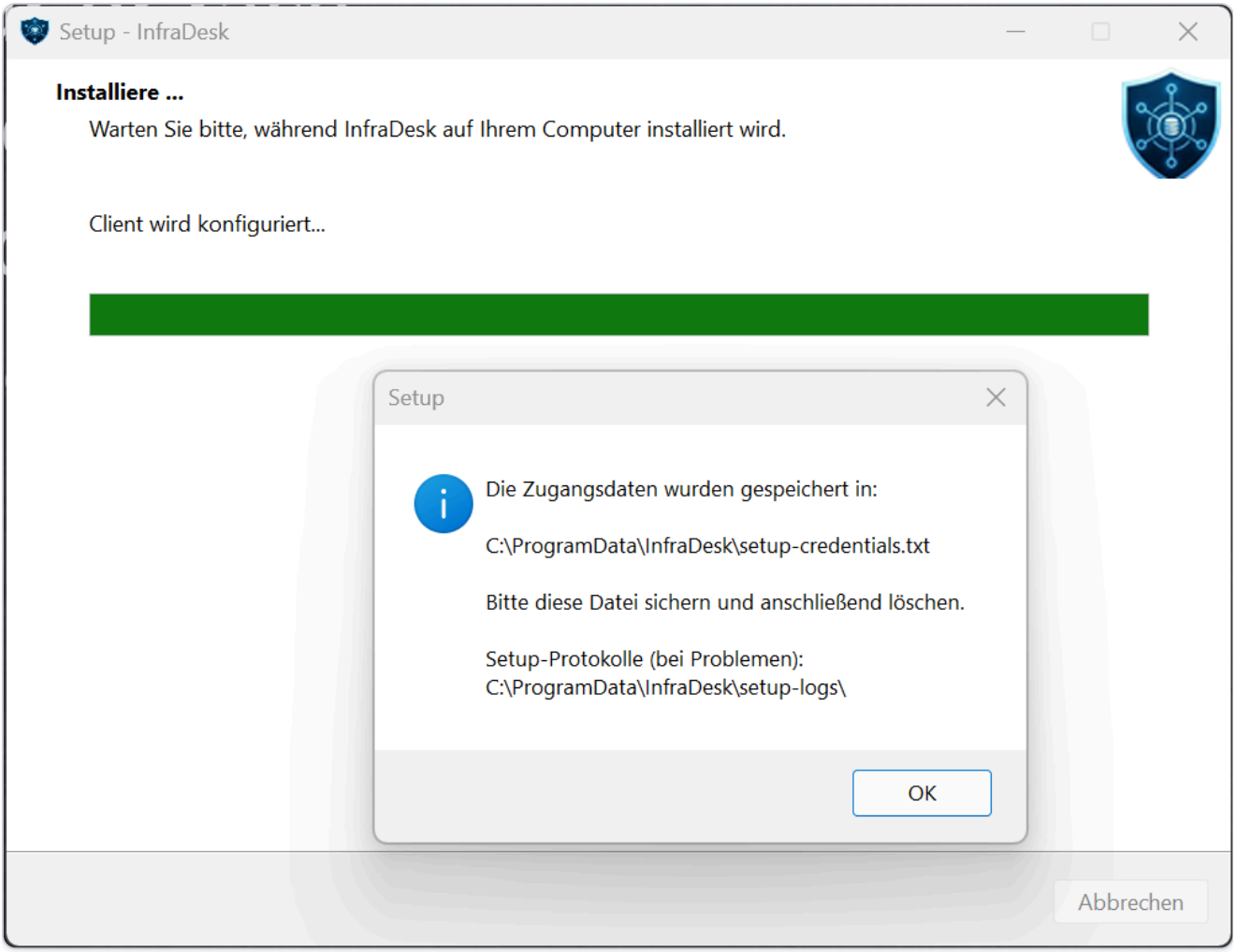

Abb. 15 – Zum Abschluss werden die **Zugangsdaten** in `C:\ProgramData\InfraDesk\setup-credentials.txt` gespeichert. Diese Datei sichern und anschließend löschen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

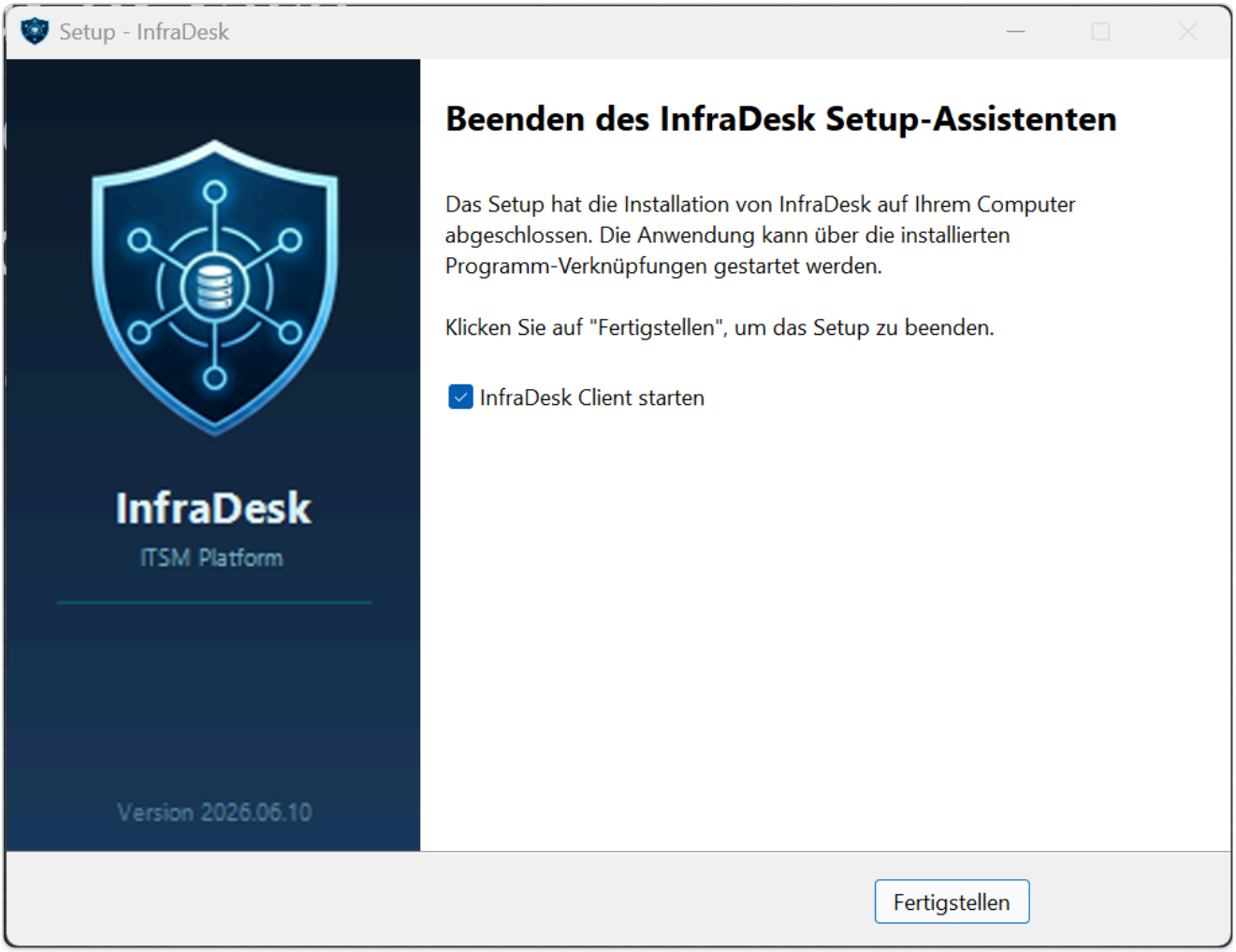

Abb. 16 – Fertig. Beim ersten Start des Clients (oder Aufruf der API-URL) beginnt die geführte Ersteinrichtung (Haupt-Mandant, Zentrale, erster Administrator – siehe ID-40).

2.3 PostgreSQL-Tuning bei der Installation

Bringt der Installer PostgreSQL **selbst** mit, stellt er den Server anschließend automatisch auf den Betrieb ein. Die Werte werden dabei **aus der Zielmaschine berechnet** – Arbeitsspeicher, Prozessorkerne und freier Platz am Datenverzeichnis –, nicht aus einer festen Liste übernommen: Ein Server mit 64 GB und eine VM mit 8 GB brauchen verschiedene Werte.

Gesetzt werden `shared_buffers`, `effective_cache_size`, `work_mem`, `maintenance_work_mem`, `max_connections`, `max_wal_size` / `min_wal_size` sowie `idle_in_transaction_session_timeout`; zusätzlich wird `pg_stat_statements` aktiviert (Grundlage für die Abfrage-Analyse). Geschrieben wird per `ALTER SYSTEM` nach `postgresql.auto.conf` – die von Ihnen gepflegte `postgresql.conf` bleibt unberührt, und jeder Wert lässt sich mit `ALTER SYSTEM RESET <name>` einzeln zurücknehmen.

> **Eine mitgebrachte PostgreSQL-Instanz wird nicht angefasst.** Läuft auf dem Server bereits ein PostgreSQL, oder haben Sie im Assistenten eine **externe Datenbank** gewählt, gehört deren Konfiguration Ihnen: Der Installer überspringt das Tuning dann vollständig und protokolliert das.

Das Protokoll des Laufs steht in `%ProgramData%\InfraDesk\setup-logs\tuning.log` – inklusive der berechneten Werte und der Werte, die der Server nach dem Neustart tatsächlich meldet.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Nachträglich lässt sich das Tuning jederzeit erneut anstoßen (das Skript ist idempotent):

```
# -DryRun rechnet nur und zeigt die Werte, ohne etwas zu schreiben
powershell -File tune-postgresql.ps1 -PgSuperPassword "<Kennwort>" -DryRun
```

> **Tipp:** Eine ausführliche Beschreibung jeder Einstellung samt **empfohlener Werte je Kundengröße** (Lizenzstufe) sowie der Client-Verbindungspools (`Database:MaxPoolSizePerTenant`) steht in ID-51.

3 Konfiguration (appsettings)

- **Connection-String** zur PostgreSQL-Datenbank.
- **Notifications:Smtp** für den E-Mail-Versand (interner Exchange, STARTTLS, FQDN, interne Root-CA).
- **CredentialVault:EncryptionKey** – 32-Byte Base64, **nie** in `appsettings.json`, sondern als Secret/Umgebungsvariable.
- **Identity:Jwt:Secret** – Signaturschlüssel der Anmelde-Token, **mindestens 32 Zeichen**. Ebenfalls **nie** in `appsettings.json`, sondern als Umgebungsvariable (siehe 3.1).
- Beim Kopieren von Updates `appsettings` **ausschließen**, um die Konfiguration zu erhalten.

3.1 Signaturschlüssel der Anmelde-Token (Pflicht)

> **Der Dienst startet seit Version 2026.07.14.1845 nicht mehr, wenn kein eigener Signaturschlüssel gesetzt ist.** Das ist beabsichtigt.

Die ausgelieferte `appsettings.json` enthält für `Identity:Jwt:Secret` lediglich einen **Platzhalter**. Bliebe dieser Wert stehen, liefe die Anwendung mit einem Schlüssel, der in **jeder** Installation identisch und damit öffentlich bekannt ist – jeder könnte sich damit ein gültiges Zugangstoken ausstellen, mit beliebiger Identität und beliebigen Rechten. Ein solcher Zustand darf nicht unbemerkt weiterlaufen; die Anwendung verweigert deshalb den Start und nennt im Ereignisprotokoll den Grund.

Für Installationen über `setup.ps1` ist nichts zu tun: Das Setup erzeugt bei der Installation automatisch einen zufälligen 48-Byte-Schlüssel und hinterlegt ihn als Maschinen-Umgebungsvariable `Identity__Jwt__Secret`. Diese bleibt bei Updates erhalten.

Manuell setzen (nur nötig, wenn die Installation nicht über `setup.ps1` erfolgte):

```
# 48 Byte Zufall → Base64 (64 Zeichen)
$bytes = New-Object byte[] 48
[System.Security.Cryptography.RandomNumberGenerator]::Create().GetBytes($bytes)
$secret = [Convert]::ToBase64String($bytes)

[System.Environment]::SetEnvironmentVariable("Identity__Jwt__Secret", $secret, "Machine")
Restart-Service InfraDesk.Api
```

> **Hinweis:** Wird der Schlüssel geändert, werden alle bestehenden Anmeldungen ungültig – die Benutzer melden sich einmal neu an.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3.2 Betrieb hinter einem Reverse-Proxy (Pflichtangabe seit 2026.08)

Steht ein Reverse-Proxy (IIS, nginx, HAProxy) vor der API, sieht diese ohne weitere Angabe **für jeden Aufruf dieselbe Adresse** – nämlich die des Proxys. Alle Schutzmechanismen, die je Adresse zählen (Anmeldeversuche, Aufrufgrenzen), teilen sich dann **eine gemeinsame Zählung für das ganze Haus**: Ein einzelner Nutzer kann damit die Anmeldung für alle anderen sperren.

> 🔑 Zugriff auf `appsettings.Production.json` des Servers und das Recht, den Dienst `InfraDesk.API` neu zu starten — nicht in Client oder Portal einstellbar.

🔒 **Pflichtreihenfolge** — die Angabe wirkt erst nach einem Neustart des Dienstes. Bis dahin zählen alle Aufrufe weiter gegen dieselbe Adresse.

1. Ermitteln Sie die Adressen aller Proxys, die vor der API stehen.
2. Tragen Sie sie in `appsettings.Production.json` ein (Beispiel unten).
3. Starten Sie den Dienst `InfraDesk.API` neu.
4. Prüfen Sie im Protokoll, dass Anmeldeversuche wieder je Ursprungsadresse gezählt werden.

✅ **Ergebnis:** Ein einzelner Nutzer kann die Anmeldung nicht mehr für das ganze Haus sperren.

```
"Hosting": {
  "TrustedProxies": [ "10.0.0.5", "10.0.0.6" ]
}
```

> **Die Liste ist bewusst leer vorbelegt.** Solange sie leer ist, werden `X-Forwarded-For`-Angaben **ignoriert**. Das ist richtig so: Würde die API sie ungeprüft glauben, könnte jeder Aufrufer eine beliebige Herkunft behaupten und die Aufrufgrenzen umgehen. Eintragen also **nur** die Adressen, von denen wirklich Ihr Proxy kommt – kein `*`, kein ganzes Netz.

Läuft die API **ohne** Proxy direkt am Netz, bleibt die Liste leer.

3.3 Wetter und amtliche Unwetterwarnungen einrichten

Wetterlage, Dreitagesvorschau, die **amtlichen Unwetterwarnungen** des Deutschen Wetterdienstes, die amtliche Warnkarte und der Wald- und Graslandfeuerindex erscheinen im **Web-Portal, im WinUI-Client und auf dem Wallboard**.

> **Achtung – seit 2026.08 stehen die Einstellungen woanders.** Der Abschnitt `Weather` gehört jetzt in die `appsettings.json` der **API** (`InfraDesk.App`), früher stand er beim **Portaldienst**. Nur die API fragt noch nach draußen; die Anzeigeprogramme holen alles von ihr. Wer `Weather:*` beim Portal gepflegt hatte, trägt die Werte um – dort werden sie **nicht mehr gelesen**, ohne Fehlermeldung.

```
"Weather": {
  "Enabled": true,
  "StationIdOverride": "",
  "LocationOverride": "",
  "CacheMinutes": 15
}
```

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Wert	Bedeutung
`Enabled`	Auf `false` setzen, wenn der Server **keinen Weg ins Internet** hat. Sonst läuft jeder Abruf in einen Zeitablauf.
`StationIdOverride`	**Leer lassen.** Die Station ergibt sich dann aus der **Postleitzahl im Mandantenprofil**. Ein Eintrag hier übersteuert sie – in einer Installation mit mehreren Mandanten für **alle** zugleich. Gedacht für den Einzelfall, in dem die abgeleitete Station danebenliegt.
`LocationOverride`	Anzeigename zur übersteuerten Station. Der DWD liefert zur Kennung **keinen Klarnamen** mit – bliebe das Feld leer, stünde auf der Seite eine nackte Zahl. Ohne Übersteuerung unbenutzt.
`CacheMinutes`	Wie lange ein Abruf gilt. Der DWD rechnet stündlich; kleinere Werte bringen keine neueren Zahlen, belasten aber eine fremde, kostenlose Schnittstelle.

Der frühere Schlüssel `StationId` wird nicht mehr gelesen. Er trug die ausgelieferte Vorgabe `10384` (Berlin-Tempelhof) und stand damit in *jeder* Installation – genau deshalb sah jeder Mandant das Berliner Wetter. Tragen Sie stattdessen die Postleitzahl im Mandantenprofil ein.

Warnungen sind gemeindescharf, das Wetter ist es nicht. Der DWD warnt nicht für Messstationen, sondern für **Warnzellen** (in aller Regel Gemeinden). Messwerte und Vorschau hängen weiterhin an der nächstgelegenen Station, die amtlichen Warnungen dagegen an Ihrer Gemeinde. Beide Angaben können daher unterschiedliche Ortsnamen tragen – das ist richtig so und keine Fehlkonfiguration.

Netzfreigabe. Benötigt wird ausgehend **HTTPS (443)** vom **API-Server** (nicht vom Portal) zu:

Ziel	Wofür	Einstellbar über
`app-prod-ws.warnwetter.de`	Messwerte und Dreitagesvorschau	fest
`api.brightsky.dev`	amtliche Warnungen, aufgelöst auf Warnzellen	`Weather:AlertsBaseUrl`
`maps.dwd.de`	amtliche Warnkarte (WMS)	`Weather:MapBaseUrl`
`opendata.dwd.de`	Wald- und Graslandfeuerindex	`Weather:OpenDataBaseUrl`

Übertragen werden dabei keine Nutzer-, Mandanten- oder Gerätedaten – nur Stationskennung bzw. Koordinaten des abgefragten Ortes.

Für den Katastrophenschutz: Bright Sky lässt sich selbst betreiben. Der Dienst bereitet die Offenen Daten des DWD als JSON auf und löst Koordinaten zu Warnzellen auf. Er ist quelloffen; wer die Abhängigkeit von einem fremden Dienst nicht tragen will, betreibt ihn im eigenen Netz und trägt die eigene Adresse unter `Weather:AlertsBaseUrl` ein.

Wer wird benachrichtigt. Bei einer **neuen** amtlichen Warnung benachrichtigt die Anwendung alle Konten mit dem Recht `weather.warning.receive` – in aller Regel über die mitgelieferte Rolle **„Wetter-Warnung“**. Sie ist ein reiner Verteiler ohne jede Befugnis und wird **zusätzlich** zur eigentlichen Rolle vergeben. Dieselbe Warnung weckt niemanden zweimal, auch wenn der DWD sie über zwei Tage hinweg wiederholt.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3.4 Windows-Anmeldung im Web-Portal (seit 2026.08)

Im Hausnetz kann sich ein bereits am Windows-Rechner angemeldeter Benutzer **ohne Kennworteingabe** im Portal anmelden. Die Anmeldemaske versucht das beim Aufruf einmal von selbst; scheitert es, steht die gewöhnliche Anmeldung mit Benutzername und Kennwort unverändert bereit.

Warum es zwei Bausteine braucht. Der Browser handelt die Windows-Anmeldung mit dem **Portal** aus, nicht mit der Schnittstelle. Diese Identität lässt sich nicht weiterreichen – dafür wäre eine eingeschränkte Kerberos-Delegierung nötig, also ein Eingriff in Ihren Verzeichnisdienst. Stattdessen prüft das Portal selbst und weist sich bei der Schnittstelle mit einem **gemeinsamen Geheimnis** aus.

Vier Angaben, die zusammen gelten. Fehlt eine, bleibt die Windows-Anmeldung im Portal aus – die Anmeldung mit Kennwort funktioniert weiterhin:

Wo	Einstellung	Bedeutung
Portal	`WindowsSso:Enabled`	Auf `true` setzen.
Portal (Dienstkonto)	–	Der Portaldienst muss unter einem **Domänenkonto** mit passendem SPN laufen, nicht unter `LocalSystem`.
Portal	`WindowsSso:TrustSecret`	Das gemeinsame Geheimnis.
API	`Identity:WindowsSso:PortalTrustSecret`	**Derselbe** Wert wie oben.
API	`Identity:WindowsSso:TrustedOrigins`	Die Adresse des Portals, z. B. `https://portal.firma.local`.

> **Die Liste `TrustedOrigins` ist bewusst leer vorbelegt, und leer bedeutet ausdrücklich „nur vom selben Rechner", nicht „von überall".** Das Geheimnis allein soll nicht genügen: Wer es erbeutet, könnte sonst von jedem Rechner im Netz eine beliebige Identität behaupten. Tragen Sie die Portaladresse ein, sobald Portal und Schnittstelle auf verschiedenen Rechnern laufen.

Das Geheimnis wird verschlüsselt abgelegt. Steht es im Klartext in der `appsettings.json`, bemängelt die Schnittstelle das beim Start im Protokoll. Verschlüsseln Sie es mit dem Maschinenschlüssel; der Wert trägt dann das Präfix `DPAPI:`.

Ohne den Schalter der Client-Anmeldung. Die Windows-Anmeldung des Portals hängt **nicht** an `Identity:WindowsSso:Enabled`. Dieser Schalter steuert das Anmeldeverfahren der Schnittstelle selbst (für den Windows-Client) und kann aus anderen Gründen aus sein, etwa hinter einem Reverse-Proxy. Der Schalter des Portalwegs ist das hinterlegte Geheimnis.

Was nicht geht. Mandantenwahl und zweiter Faktor führen weiterhin über die gewöhnliche Anmeldung – beides gehört dorthin. Und das Notfallkonto ist über die Windows-Anmeldung **nicht** erreichbar; das ist Absicht.

3.5 Sitzungsdauer (seit 2026.08)

Wie lange eine Anmeldung hält, steht an **zwei** Stellen — je einmal für die Schnittstelle und für das Web-Portal. Beide Fristen sind **gleitend**: Jeder Zugriff setzt sie neu, sie laufen also erst nach so langer **Ruhe** ab. Sie sind keine Höchstdauer; ein geöffneter Windows-Client hält seine Sitzung durch seine regelmäßigen Abrufe von allein offen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Anwendung	Einstellung	Vorgabe
Schnittstelle (Windows-Client, Mobil)	`Identity:Jwt:RefreshTokenExpiryHours`	12 Stunden
Web-Portal	`Session:LifetimeMinutes`	30 Minuten

Der große Unterschied ist Absicht. Am Windows-Client arbeitet eine benannte Person an einem Arbeitsplatz, den sie sperrt; das Portal steht Endanwendern offen und läuft auch an gemeinsam genutzten Rechnern, etwa im Bürgerbüro oder in der Werkstatt. Eine halbe Stunde Ruhe ist dort die passendere Grenze. Wer längere Portal-Sitzungen braucht, trägt einen höheren Wert ein — `480` wären die früheren acht Stunden.

> ⚠️ **Wo die Werte der Schnittstelle einzutragen sind.** Auf einem installierten Server gelten die **Maschinen-Umgebungsvariablen** `Identity__Jwt__RefreshTokenExpiryHours` und `Identity__Jwt__AccessTokenExpiryMinutes`; der Installer setzt sie. Sie haben Vorrang vor `appsettings.Production.json` — und sind zugleich der einzige Weg, der eine **bestehende** Installation erreicht, denn Updates kopieren `appsettings*.json` bewusst nicht mit (Abschnitt 3, letzter Punkt).

Der Wert des Portals steht in dessen `appsettings.json`; sie überlebt Updates aus demselben Grund. Nach einer Änderung den jeweiligen Dienst neu starten (`InfraDesk.API` bzw. `InfraDesk.Portal`).

> ⚠️ `Session:LifetimeMinutes` **auf 0 oder darunter ist keine Laufzeit, sondern ein Tippfehler** — er würde jeden Anwender sofort abmelden. Das Portal nimmt in diesem Fall die Vorgabe und schreibt eine Warnung ins Protokoll.

Das kurzlebige Zugriffstoken (`Identity:Jwt:AccessTokenExpiryMinutes`, 15 Minuten) ist **nicht** die Sitzungsdauer: Es wird im Hintergrund erneuert, ohne dass jemand es merkt. Es hochzusetzen verlängert nur die Zeit, bis entzogene Rechte oder eine gesperrte Kennung greifen — für die Anwender ändert es nichts.

4 Datenbank-Migrationen

Migrationen werden mit der Auslieferung bereitgestellt und beim Update angewendet.

> **Achtung:** Neue Tabellen müssen der App-Rolle gehören. Wird eine Migration versehentlich als `postgres` gegen die echte Datenbank getestet, gehören die Tabellen `postgres`, und die als `infradesk_app` laufende API erhält **42501 / DbUpdateException**. Abhilfe: `ALTER TABLE … OWNER TO infradesk_app`.

4.1 Mandanten-Provisionierung (eigene Datenbank je Mandant)

Legt ein SuperAdmin über **Einstellungen → Mandanten** einen neuen Mandanten an, erzeugt die Anwendung dafür eine **eigene Datenbank** (`infradesk_<slug>`). Dazu muss die verwendete Datenbank-Rolle das Recht `CREATEDB` besitzen.

Bei einer **Neuinstallation** wird das bereits eingerichtet: Setup/Installer legt die App-Rolle `infradesk_app` mit `CREATEDB` an – die Mandanten-Provisionierung funktioniert also **direkt**.

Nur bei **älteren Bestandsinstallationen** kann das Recht fehlen; dann bricht das Anlegen mit **„42501: keine Berechtigung, um Datenbank zu erzeugen“** ab. Abhilfe (als PostgreSQL-Superuser):

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
ALTER ROLE infradesk_app CREATEDB;
```

> **Härtungs-Alternative (KRITIS/BSI):** Statt der App-Rolle `CREATEDB` zu geben, kann in `appsettings.json` unter `ConnectionStrings` ein Eintrag `OrdivisMaintenance` mit einer eigenen, nur zum Anlegen von Datenbanken berechtigten Rolle hinterlegt werden. Das Provisioning nutzt diese Verbindung bevorzugt; die App-Rolle bleibt dann ohne `CREATEDB` (Least Privilege).

Das **Sicherungs-Verzeichnis** wird im Setup abgefragt (Standard nicht unter `C:\ProgramData`, sondern ein dediziertes Datenverzeichnis, z. B. `D:\Ordivis Platform\Backups`) und sowohl an die geplante Aufgabe (`backup.ps1 -BackupDir`) als auch an die Anwendung (`Backup:Directory`) übergeben.

5 In-Place-Update & Versionierung

- Die App-Version folgt dem Schema **JJJJ.MM.TT.HHMM**.
- Update: Dienst stoppen → Dateien kopieren (ohne `appsettings`) → Migrationen → Dienst starten.

> **Achtung:** Der erste `Start-Service` nach dem Kopieren der API scheitert häufig am SCM-Timeout/Kaltstart – **einfach erneut starten**.

> **Achtung (IPAM/Deploy-Falle):** Stoppt der Dienst nicht innerhalb von 60 s, bleiben geladene DLLs gesperrt und werden nicht überschrieben (Symptom: 404 statt 401). Dann den Prozess **elevated per** `taskkill` beenden und erneut deployen.

> **Hinweis (Client/WinUI):** Zum Bauen des Clients `/t:Build` verwenden (nicht `/t:Publish`) – ein Publish entfernt die PRI-Ressourcen, wodurch der Client nach dem Splash abstürzt (XamlParseException).

5.1 Delta-Update (signiert, client-gesteuert)

Ergänzend zum manuellen In-Place-Update gibt es das **Delta-Update-Verfahren**: Statt das gesamte Setup zu übertragen, wird je Version ein **signiertes Release-Manifest** ausgeliefert, und das Ziel lädt nur die **abweichenden** Dateien.

- **Release-Manifest:** je Produktversion ein mit **ECDSA (ES256)** signiertes `manifest.json` mit der Datei-Liste (Pfad, SHA-256, Größe) pro Komponente (API, Portal, Client, Wallboard und seit 2026.08.12.1127 der Aktualisierungsdienst des Arbeitsplatzes, siehe 5.1.2). Der Hersteller erzeugt es mit dem Werkzeug `tools/InfraDesk.UpdateManifestGenerator` aus den publizierten Dist-Ordnern; der private Signaturschlüssel bleibt ausschließlich beim Hersteller, der **öffentliche** Schlüssel ist im Produkt eingebettet. Ein Manifest ohne gültige Signatur wird **nie** angewandt.
- **Update-Quelle:** `https://…/updates/<version>/` mit `manifest.json` und **inhaltsadressierter Blob-Ablage** (`<component>/<hash>`) – identische Dateien über Versionen hinweg teilen sich einen Blob.
- **Delta am Ziel:** Das Produkt vergleicht die installierten Datei-Hashes mit dem Manifest und lädt nur die fehlenden/geänderten Blobs (jeweils gegen ihren Hash verifiziert) in ein **Staging-Verzeichnis** (`%ProgramData%\InfraDesk\update-staging\<version>`) – **nicht-destruktiv**, die laufende Installation bleibt unangetastet.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Anstoßen (SystemAdmin):** `GET /api/v1/system/update/delta-status` zeigt den signatur-geprüften Umfang (MB/Dateien); der Client zeigt ihn unter *Einstellungen ▸ Updates* hinter „Nach Updates suchen". `POST /api/v1/system/update/stage/<version>` lädt die Blobs. `activation.manage` zum Lesen, `system.config.manage` zum Anstoßen.
- **Anwenden & Rollback (entkoppelter Updater):** Das eigentliche Übernehmen aus dem Staging läuft seit Version **2026.07.15.1738 nicht mehr in der laufenden API**, sondern in einem eigenständigen `InfraDesk.Updater` (Bootstrapper), der den Stopp von API und Client übersteht. Er wendet **alle lokalen Komponenten** an – je Komponente: Dienst **sauber stoppen** (`sc stop` → auf STOPPED warten → nur zur Not `taskkill /F` gegen DLL-Locks) → **Snapshot** anlegen → Dateien übernehmen (mit Copy-Retry) → **starten** → **Health**. Schlägt ein Schritt fehl, wird der Snapshot **zurückgespielt** (Vorversion). Vor dem gesamten Lauf wird ein **DB-Backup** angestoßen. Das geprüfte Ergebnis je Komponente landet in `%ProgramData%\InfraDesk\update-status.json` (der Client zeigt es beim nächsten Start).
- **Log:** je Host `%ProgramData%\InfraDesk\update-logs\` plus zentrales Update-Log (DB, Schema `system`) und Audit-Ereignis `system.update.performed`.

> **Hinweis (Reihenfolge & verteilte Hosts):** Der Orchestrator im Backend stellt einen Gesamtplan über alle betroffenen Komponenten zusammen und übergibt ihn dem entkoppelten Updater. Reihenfolge: **Dienste zuerst** (Portal, dann API – die API zuletzt unter den Diensten, damit sie ihre HTTP-Antwort noch senden kann), der **Client ganz zuletzt** (er wird erst nach gesunder API neu gestartet, und nur, wenn er lief). Liegen API und Portal auf **getrennten Hosts**, wird die jeweils fremde Komponente über deren lokalen **Update-Agent** aktualisiert.

5.1.1 ARBEITSPLATZ-CLIENTS (GETRENNTE INSTALLATION)

Der Orchestrator plant ausschließlich Komponenten, deren Verzeichnis auf dem **Server-Host** liegt. Ein Client, der auf einem eigenen Arbeitsplatz-PC installiert ist, wird davon **nicht erfasst** – auch dann nicht, wenn das Update von genau diesem Client ausgelöst wurde. Dafür sorgt seit **2026.07.22.1615** ein eigener Dienst auf dem Arbeitsplatz.

Aktualisierungsdienst `InfraDesk.ClientUpdate`

Bei einer **reinen Client-Installation** richtet das Setup diesen Windows-Dienst ein (verzögerter Automatikstart, Konto `LocalSystem`). Auf einem Host, auf dem auch die API läuft, wird er **nicht** registriert – dort erledigt der Orchestrator die Client-Kopie bereits.

- **Zeitplan:** Der Dienst fragt standardmäßig alle **4 Stunden** beim eigenen Server nach der Zielversion. Er ist von außen nicht erreichbar (kein eingehender Port, keine Firewall-Regel nötig).
- **Zielversion** ist die **laufende Version des eigenen Servers**, nicht die neueste auf der Produkt-Website. So bleiben Arbeitsplatz und Backend im Gleichschritt; angehoben wird der Arbeitsplatz erst, wenn der Server selbst aktualisiert wurde.
- **Erst laden, dann fragen:** Manifest (**ES256**) und Blobs (**SHA-256** je Blob) werden vollständig geprüft heruntergeladen, bevor irgendjemand gefragt wird. Die Installation bleibt dabei unberührt.
- **Läuft der Client nicht**, wird sofort und ohne Rückfrage aktualisiert – der häufigste Fall.
- **Läuft der Client**, fragt er den Benutzer: *jetzt aktualisieren* oder *um 1 / 4 / 16 Stunden verschieben*. **Beendet wird nur nach Zustimmung.** Schließt der Benutzer Ordivis Platform zwischenzeitlich von sich aus, nutzt der Dienst diesen Moment.
- **Keine UAC-Rückfrage:** Der Dienst läuft als `LocalSystem` und darf unter `C:\Programme` schreiben. Anwender **ohne lokale Administratorrechte** können ihren Client damit aktualisieren.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **Nachprüfung:** Nach dem Dateitausch liest der Dienst die Version **an der EXE** erneut aus. Nur wenn sie der Zielversion entspricht, gilt der Lauf als erfolgreich; andernfalls wird der Snapshot zurückgespielt. Das Ergebnis meldet der Client beim nächsten Start.
- **Server-Adresse:** Der Dienst ermittelt sie aus denselben Quellen wie der Client, in dieser Reihenfolge: 1. `%ProgramData%\InfraDesk\connections.json` – das **Leitprofil** aus dem Verbindungsverzeichnis (siehe Abschnitt *Mehrere Installationen aus einem Client*). Diese Quelle ist maschinenweit und eindeutig; erreicht ein Arbeitsplatz mehrere getrennte Installationen, bestimmt sie, auf welchen Stand er gehoben wird. 2. `%LocalAppData%\InfraDesk\connection.json` – die im Client unter *Verbindungseinstellungen* zuletzt gewählte Adresse. Sie liegt im Benutzerprofil; der Dienst läuft als `LocalSystem` und durchsucht deshalb die Profile und nimmt das zuletzt geschriebene. 3. `<Installationsverzeichnis>\client\server-url.txt` – die Eingabe aus dem Setup. 4. `<Installationsverzeichnis>\client\debugConfig.json` – mitgelieferte Vorgabe (`http://localhost:5000`); auf einem eigenen Arbeitsplatz-PC ist sie fast immer falsch.

Welche Quelle gewonnen hat, steht im Protokoll. Fehlen die Punkte 1 bis 3, fragt der Dienst `localhost` und findet dort nichts – dann genügt es, `server-url.txt` mit der richtigen Adresse anzulegen und den Dienst neu zu starten. - **Protokoll:** `C:\ProgramData\InfraDesk\Logs\clientupdate-JJJJMMTT.log` sowie das Windows-Ereignisprotokoll (Quelle `InfraDesk.ClientUpdate`). Es nennt die verwendete Server-Adresse samt Herkunft, den Wartezustand vor dem Schließen des Clients und das Ergebnis der Nachprüfung.

> **Nach dem Update starten Sie Ordivis Platform wieder selbst** – der Dienst startet den Client bewusst nicht automatisch neu.

Rückfallweg ohne Dienst: Ist der Dienst nicht eingerichtet, nutzt der Client den mitgelieferten `InfraDesk.Updater.exe` (gleiche Prüfungen, gleicher Snapshot-Mechanismus). Liegt die Installation unter `C:\Programme`, verlangt dieser Weg eine **UAC-Rückfrage**; wird sie abgelehnt, bricht die Aktualisierung mit einer klaren Meldung ab.

> **Einmalig:** Clients, die **vor** diesem Release installiert wurden, bringen den Aktualisierungsdienst noch nicht mit. Sie müssen **einmal** über die Setup-Datei aktualisiert werden (Installationstyp „Client-Installation"); ab dann läuft die Aktualisierung von selbst.

Mehrere Installationen aus einem Client

Nicht jede Einrichtung führt alles auf einem Server. Wer mehrere getrennte Ordivis-Installationen betreibt – etwa Hauptverwaltung und Stadtwerke –, kann sie aus **einem** Client erreichen und auf der Anmeldemaske umschalten.

> ⚠️ **Nicht zu verwechseln mit Mandantenfähigkeit.** Dort teilen sich mehrere Mandanten *eine* Installation (ein Server, eine Registry, je Mandant eine Datenbank). Hier erreicht *ein Client mehrere getrennte Installationen* – jede mit eigenem Server, eigener Registry, eigener Version. Der Mandantenwechsler im Programm bleibt davon unberührt.

Fehlt die Datei, ändert sich nichts. Bestandsinstallationen laufen unverändert weiter, die Auswahl bleibt unsichtbar. Sie erscheint erst ab **zwei** Einträgen.

AUFBAU DER DATEI

Die Liste ist maschinenweit und liegt in `%ProgramData%\InfraDesk\connections.json`:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
{
  "version": 1,
  "allowCustom": false,
  "connections": [
    { "id": "hv",  "name": "Hauptverwaltung", "apiBaseUrl": "https://ordivis.stadt.local:5001",
      "ignoreCertificateErrors": false, "useWindowsSso": true, "primary": true },
    { "id": "swk", "name": "Stadtwerke",      "apiBaseUrl": "https://sw.stadt.local:5001",
      "ignoreCertificateErrors": true,  "useWindowsSso": false }
  ]
}
```

Feld	Bedeutung
`id`	Kurze, gleichbleibende Kennung. Sie hält die Auswahl fest, auch wenn sich die Adresse ändert.
`name`	So erscheint die Installation auf der Anmeldemaske.
`apiBaseUrl`	Vollständige Adresse der API mit Schema und Port.
`ignoreCertificateErrors`	Zertifikatoption **dieser** Verbindung. Vorher galt sie für den ganzen Arbeitsplatz – bei einem Server mit gültigem und einem mit selbstsigniertem Zertifikat war das falsch.
`useWindowsSso`	Windows-Anmeldung **dieser** Verbindung. Produktivserver automatisch, Testserver nicht.
`primary`	Das **Leitprofil**. Genau ein Eintrag; fehlt die Marke, gilt der erste.
`allowCustom`	Ob daneben eine eigene Adresse zulässig ist. Vorgabe **aus** – die Liste gibt der Administrator vor.

DAS LEITPROFIL BESTIMMT DIE VERSION

Der Arbeitsplatz-Dienst richtet sich nach dem Leitprofil, nicht nach der zuletzt gewählten Verbindung. Ohne diese Festlegung zöge er den Arbeitsplatz auf die Version des **neuesten** Servers, mit dem er je verbunden war – und der Client spräche danach gegen eine ältere API, die er nicht kennt. Aus demselben Grund bietet der Veraltet-Hinweis im Client den Tausch **nur beim Leitprofil** an; bei den übrigen Verbindungen steht der Hinweis ohne Schaltfläche.

DATEIRECHTE

Ohne Schreibschutz gegen normale Benutzer wäre die Vorgabe eine Behauptung. Das Setup setzt die Rechte (Administratoren und `SYSTEM` Vollzugriff, Benutzer nur Lesen), schaltet die Vererbung ab und **liest sie zurück**. Auf einem Datenlaufwerk erbt der `ProgramData`-Zweig die Rechte nämlich nicht von selbst.

Prüfen:

```
Get-Acl "$env:ProgramData\InfraDesk\connections.json" | Format-List
```

WENN EINE FALSCHE VORGABE ALLE AUSSPERRT

Eine falsche Adresse in der Liste kann jeden aussperren, auch den Administrator. Es gibt zwei Wege zurück, beide vor Ort:

1. Die Datei auf dem Arbeitsplatz berichtigen (Administratorrechte nötig).
2. Das Setup erneut laufen lassen. ⚠️ Eine **bestehende** `connections.json` wird dabei **nicht** überschrieben – sie ist der gepflegte Bestand. Zum Zurücksetzen die Datei vorher löschen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Wer die Tür grundsätzlich offen halten will, setzt `allowCustom` auf `true`; dann bleibt der Verbindungsdialog änderbar.

WAS DER WECHSEL VERWIRFT

Ein Wechsel ist kein Abmelden, räumt aber genauso auf: Sitzung, Live-Kanal, Rechte, Statusabfrage und die mandantenbezogenen Zwischenspeicher werden verworfen, und die Maske springt auf Schritt 1 zurück. Die abgelegte Windows-Anmeldung bleibt liegen – sie gehört seit diesem Release **je Verbindung** in eine eigene Datei, damit ein Erneuerungstoken niemals an den falschen Server geht.

Nicht angefasst werden gespeicherte Ticket-Ansichten und CMDB-Filter: Sie speichern Text, keine serverseitigen Kennungen, und sind serverübergreifend höchstens unpassend, nie falsch.

ANDROID-APP

Auf Android gibt es keinen `ProgramData`-Zweig. Die Liste kommt dort aus zwei Quellen, in dieser Reihenfolge:

1. **Android Managed Configuration** aus der MDM-Lösung – der richtige Weg für verwaltete Geräte, weil der Administrator sie ohne neuen Build pflegt. Die Felder erscheinen in der MDM-Konsole.
2. Die eingebettete `appsettings.json`, Abschnitt `Api:Connections` – sie kommt mit der verteilten Anwendung.

WEB-PORTAL

Das Portal bekommt bewusst **keine** Auswahl, sondern eine Linkliste. Es ist Blazor Server und legt seine API-Adresse beim Prozessstart einmalig fest; ein Portalprozess, der mit mehreren Schnittstellen spräche, wäre ein anderes Produkt. Die Liste steht in der `appsettings.json` unter `OtherInstallations` und führt in das jeweils eigene Portal der anderen Installation. Leere Liste = kein sichtbarer Unterschied.

5.1.2 BESTEHENDE ARBEITSPLÄTZE EINMALIG ANSTOSSEN (SEIT 2026.08.12.1127)

Bis zu dieser Fassung war der Aktualisierungsdienst die einzige ausgelieferte Komponente **ohne eigenen Update-Weg**: Das signierte Manifest kannte nur API, Portal, Client und Wallboard. Der Dienst hielt den ganzen Arbeitsplatz aktuell und blieb dabei selbst auf dem Stand seiner Installation stehen. Seit **2026.08.12.1127** ist `clientupdate` die **fünfte Komponente** im Manifest: Der Dienst prüft je Zyklus auch seine eigene Version am Artefakt und übergibt den Tausch dem Bootstrapper.

Diese Fähigkeit kann sich nicht selbst ausliefern. Ein Dienst ohne Selbstupdate-Code holt seinen Nachfolger prinzipbedingt nicht über das Delta – er kennt den Weg dorthin ja gerade nicht. Bereits eingerichtete Arbeitsplätze brauchen deshalb **genau einmal** einen Anstoß von außen. Ohne ihn bleibt der Aktualisierungsdienst auf altem Stand stehen, während er den Client weiter zuverlässig anhebt – ein Zustand, der von außen unauffällig aussieht. Danach hält der Dienst sich selbst aktuell, und der Handgriff wiederholt sich nie.

Dafür liegt `update-clientupdate.ps1` auf dem **Server** bereit, im Unterverzeichnis `scripts` Ihrer Installation – also neben `backup.ps1`. **Quelle** für die neuen Dateien ist `clientupdate` derselben Server-Installation: Eine Server-Installation trägt die Dateien des Aktualisierungsdienstes mit, registriert ihn dort aber bewusst nicht. Damit ist Ihr Server zugleich die Verteilstelle, und Sie müssen nichts zusätzlich bereitlegen.

Angenommen, die Installation liegt unter `D:\Program Files\Ordivis Platform`:

```
# Erst zeigen, was geschaehe – ohne eine Datei anzufassen
powershell -ExecutionPolicy Bypass -File "D:\Program Files\Ordivis Platform\scripts\update-clientup
```

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
# Dann ausfuehren, in einer Eingabeaufforderung MIT Administratorrechten
powershell -ExecutionPolicy Bypass -File "D:\Program Files\Ordivis Platform\scripts\update-clientup
```

Auf einem **entfernten** Arbeitsplatz geben Sie beides als Netzwerkpfad an – geben Sie `scripts` und `clientupdate` dazu auf dem Server frei, oder kopieren Sie beide Verzeichnisse einmal auf eine vorhandene Freigabe.

- **Bewusst nicht der Installer.** Das Skript fasst ausschließlich `<Installationsverzeichnis>\clientupdate` an: Dienst stoppen, Dateien tauschen, Dienst starten. **Keine Datenbank, kein PostgreSQL, kein** `%ProgramData%` – nichts von dem, was ein Installer-Lauf auf einer produktiven Anlage sonst berührt (Datenbank-Provisionierung, Bereinigung von Altbeständen). `appsettings*.json` bleibt liegen; eine angepasste Server-Adresse oder ein geändertes Prüfintervall überleben den Lauf.
- **Administratorrechte sind Pflicht.** Ein als `LocalSystem` laufender Dienst lässt sich nur erhöht stoppen, und unter `C:\Programme` darf ein gewöhnliches Konto nicht schreiben. Fehlen die Rechte, bricht das Skript **vor** der ersten Änderung ab, nicht auf halbem Weg.
- `-WhatIf` **zeigt den Lauf vorab:** installierte Version, Version der Quelle und Zielverzeichnis, ohne etwas zu ändern. Das Zielverzeichnis stammt aus der Dienst-Registrierung – geraten wird nichts. Mit `-InstallDir` lässt es sich vorgeben.
- **Der Nachweis steht am Artefakt, nicht am Exit-Code.** Nach dem Tausch liest das Skript die Version an der `InfraDesk.ClientUpdateService.exe` erneut aus und prüft, dass der Dienst wieder `Running` meldet. Stimmt eines von beidem nicht, endet der Lauf mit **Exit-Code 1** – wichtig, wenn Sie über mehrere Arbeitsplätze schleifen oder das Skript aus einer Verteilsoftware aufrufen.
- **Liegt bereits der neue Stand an**, meldet das Skript das und tut nichts. Ein versehentlicher zweiter Lauf schadet also nicht.

Welcher Handgriff gilt für Sie? Prüfen Sie auf dem Arbeitsplatz, was vorhanden ist – die erste Zeile nennt das Installationsverzeichnis, die zweite sagt, ob die Programmdateien liegen, die dritte, ob der Dienst eingerichtet ist:

```
$app = (Get-ItemProperty 'HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\*' | Where-Obje
```

Befund	Handgriff
Dienst vorhanden, aber älter als **2026.08.12.1127**	`update-clientupdate.ps1` – dieser Abschnitt
Dateien liegen, **Dienst fehlt**	`register-client-update-service.ps1`, ebenfalls in `scripts` – der Absatz unter dieser Tabelle
Dateien fehlen (Installation vor 2026.07.22.1615)	Setup-Datei, Installationstyp „Client-Installation"

Der mittlere Fall braucht nur die Registrierung, weil die Programmdateien bei jeder Client-Installation mitkommen – auch bei einer, die den Dienst damals noch nicht eingerichtet hat. Das Skript verlangt das Installationsverzeichnis und legt den Dienst genauso an wie das Setup (verzögerter Automatikstart, `LocalSystem`, Neustart nach Fehlern); ist er bereits vorhanden, wird er zuvor sauber entfernt:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
# In einer Eingabeaufforderung MIT Administratorrechten
powershell -ExecutionPolicy Bypass -File "D:\Program Files\Ordivis Platform\scripts\register-client
```

`-InstallDir` ist das Verzeichnis auf dem **Arbeitsplatz** – auf einem Laptop üblicherweise unter `C:`, während der Server unter `D:` liegt. Das Skript bricht ab, statt einen Dienst anzulegen, der beim Start sofort scheitern würde, wenn dort keine `clientupdate\InfraDesk.ClientUpdateService.exe` liegt.

5.2 Update-Center & Auto-Update-Policy

- **3-Stufen-Policy** (persistiert in `SystemState`; Lesen `activation.manage`, Ändern `system.config.manage`): **Aus** / **Nur benachrichtigen** / **Vollautomatisch**. Bei Vollautomatik wendet der `ScheduledUpdateWorker` das signaturgeprüfte Delta selbsttätig im konfigurierten **Wartungsfenster** (Wochentage + Uhrzeit, 1-h-Fenster) an – mit Pre-Update-Backup, Health-Check und automatischem Rollback wie bei manueller Ausführung; der Lauf ist je Zielversion **dedupliziert** (`update.last_auto_applied_version`). Endpunkte `GET/PUT /api/v1/system/update/policy`.
- **Changelog-Quelle:** Die Website führt neben `version.json` eine fortgeschriebene `changelog.json` (je Release: `version`, `date`, `title`, `mandatory`, `components`, `changes` mit `category`/`text`). `GET /api/v1/system/changelog` liefert sie aufbereitet (10 min gecacht); der Publish-Dienst überträgt `changelog.json` wie `version.json` bei jedem Release.
- **Komponentenansicht:** je Komponente (API/Portal/Client/Wallboard) installierte Version, verfügbare Zielversion und „betroffen"-Kennzeichnung aus dem Delta-Plan – direkt im Client-Update-Center.

> **Zwei Rechner, eine Liste – auf die Beschriftung achten:** Alle Komponenten dieser Ansicht liegen auf dem **Server-Host**; die Zeile „Client (Kopie auf dem Server)" meint die Client-Installation *neben der API*, nicht Ihren Arbeitsplatz. Die eigene Installation steht getrennt darüber als **„Dieser Arbeitsplatz"**. Bei getrennten Installationen zeigen beide Zeilen zu Recht unterschiedliche Versionen.

6 Sicherung & Wiederherstellung

Die Datenbank wird server-seitig durch die geplante Windows-Aufgabe `InfraDesk-DailyBackup` gesichert (`pg_dump`). Ziel-Verzeichnis, Zeitplan und Aufbewahrung stellen Sie als **SuperAdmin** unter *Einstellungen → Backup* ein; von dort lässt sich eine Sicherung auch sofort anstoßen und eine vorhandene Sicherung wiederherstellen.

6.1 Verschlüsselte Sicherungen (7z, AES-256)

> 💻 **Nur im Client:** *Fußbereich ▸ Einstellungen ▸ Backup* · 🔑 `system.backup.manage`

Ist ein **Sicherungs-Kennwort** hinterlegt, werden die Sicherungen als **AES-256-verschlüsselte 7z-Archive** (`ordivis_<Zeitstempel>.7z`) abgelegt; der unverschlüsselte Zwischenstand wird nach dem Verschlüsseln gelöscht. Ohne Kennwort bleibt es bei einer unverschlüsselten `.dump`-Datei (die Oberfläche weist darauf hin).

🔒 **Pflichtreihenfolge** — hinterlegen Sie das Kennwort **an einem zweiten, sicheren Ort**, bevor Sie es hier speichern. Ohne das Kennwort ist eine verschlüsselte Sicherung wertlos, und es gibt keinen Weg daran vorbei.

1. Legen Sie das gewählte Kennwort im Kennwortspeicher Ihres Hauses ab.
2. Öffnen Sie *Einstellungen ▸ Backup*.
3. Tragen Sie das **Sicherungs-Kennwort** ein.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

4. Speichern Sie.
5. Warten Sie den nächsten Sicherungslauf ab und prüfen Sie, dass eine `.7z`-Datei entsteht.

✅ **Ergebnis:** Ab der nächsten Sicherung entstehen verschlüsselte Archive.

- Das Kennwort gilt **zentral** für die Installation. Ein **leeres** Feld beim Speichern lässt ein bereits hinterlegtes Kennwort unverändert; ein neuer Wert ersetzt es und wirkt ab der nächsten Sicherung.
- **Wiederherstellung:** Beim Zurückspielen eines verschlüsselten Archivs wird das Kennwort **abgefragt** (nicht aus der Konfiguration übernommen). So lässt sich auch eine ältere Sicherung mit abweichendem Kennwort zurückspielen. Unverschlüsselte `.dump`-Sicherungen werden weiterhin ohne Abfrage wiederhergestellt.

Achtung: Ohne das Kennwort lässt sich eine verschlüsselte Sicherung **nicht** wiederherstellen – es gibt keine Hintertür. Bewahren Sie das Kennwort sicher und **getrennt von den Sicherungen** auf.

Wichtig: Das Sicherungs-Kennwort wird **niemals im Klartext** abgelegt. Es wird mit **Windows-DPAPI** verschlüsselt in `operations.json` gespeichert und lässt sich nur im Dienst-/Aufgabenkontext (SYSTEM) auf **diesem** Server entschlüsseln. Wird die Datei kopiert oder von einem anderen Konto gelesen, bleibt das Kennwort unbrauchbar.

6.2 Weitere Sicherung & Wiederherstellung

- **Konfiguration & Secrets:** `appsettings` (ohne Geheimnisse) und den **Vault-Schlüssel** sicher und getrennt aufbewahren.
- **Wiederherstellung:** Datenbank zurücksichern, Dienste konfigurieren, Vault-Schlüssel bereitstellen.

Achtung: Ohne den originalen **Vault-Schlüssel** lassen sich verschlüsselte Zugangsdaten (Discovery, E-Mail) nach einer Wiederherstellung nicht entschlüsseln und müssen neu erfasst werden.

6.3 Anhänge liegen in der Datenbank

Anhänge – das Bildschirmfoto an einem Vorgang, die archivierte Roh-E-Mail, ein Dokument im BCM – werden **in der Datenbank** gespeichert. Sie sind damit von der Sicherung aus 6.1 **mit erfasst**; ein zweiter Sicherungsweg für ein Dateiverzeichnis entfällt, und eine zurückgespielte Datenbank bringt ihre Anhänge mit.

Achtung: Bis einschließlich **2026.08.12** war das anders. Die Anhänge lagen als Dateien im **Temp-Verzeichnis des Dienstkontos** – ein Ablageort, den die Datenträgerbereinigung leert, den `pg_dump` nicht erfasst und der sich mit dem Dienstkonto verschiebt. Die Zeile in der Datenbank überlebte das jedes Mal, der Inhalt nicht. Prüfen Sie nach dem Update das Protokoll der Übernahme, **bevor** Sie sich auf eine ältere Sicherung verlassen: Was dort schon fehlte, kann keine Übernahme zurückholen.

So prüfen Sie die Übernahme nach dem Update:

🔑 Lesezugriff auf das Serverprotokoll — nicht in Client oder Portal sichtbar.

🔒 **Pflichtreihenfolge** — prüfen Sie das Protokoll, **bevor** Sie sich auf eine ältere Sicherung verlassen. Was dort schon fehlte, kann keine Übernahme zurückholen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

1. Starten Sie den Dienst nach dem Update und warten Sie den ersten Lauf ab.
2. Öffnen Sie das Protokoll (Kapitel 10).
3. Lesen Sie den Eintrag zur Anhang-Übernahme.

✅ **Ergebnis:** Sie wissen, wie viele Anhänge übernommen wurden und welche gefehlt haben.

Die Übernahme läuft **je Datenbank einmal**, danach ist nichts mehr zu tun. Das Protokoll sagt, was passiert ist:

```
Anhänge: 30 Altbestand-Dateien werden in die Datenbank übernommen. Fundorte: …
Anhänge: 30 Altbestand-Dateien vollständig übernommen.
```

Gesucht wird der Reihe nach in `Attachments:RootPath` (falls gesetzt), in `Attachments:LegacyRootPaths`, im Temp-Verzeichnis des laufenden Dienstkontos sowie in `%SystemRoot%\SystemTemp` und `%SystemRoot%\Temp` – jeweils im Unterverzeichnis `infradesk-attachments`. Damit wird auch der Fall gefunden, dass der Dienst inzwischen unter einem anderen Konto läuft als beim Hochladen.

Diese Zusagen gelten für den Lauf:

- **Nichts wird gelöscht.** Die Altdateien bleiben liegen. Sie sind das Sicherheitsnetz, solange niemand die Übernahme geprüft hat; aufräumen können Sie danach von Hand.
- **Eine nicht gefundene Datei hält nichts auf.** Sie wird als Warnung gemeldet, die übrigen Anhänge werden trotzdem übernommen, und der Start bricht nicht ab.
- **Ein Fehlschlag ist wiederholbar.** Wo die Altdatei fehlte, behält die Zeile ihren Pfad. Tragen Sie den tatsächlichen Ablageort unter `Attachments:LegacyRootPaths` nach (mehrere durch **Semikolon** getrennt) und starten Sie den Dienst neu – der nächste Lauf holt sie nach.

```
{ "Attachments": { "LegacyRootPaths": "E:\\Gesichert\\infradesk-attachments;F:\\Alt\\infradesk-atta
```

So zählen Sie jederzeit nach, was noch aussteht (0 = vollständig übernommen):

```
SELECT count(*) FROM discovery.attachments WHERE storage_path IS NOT NULL;
```

7 Sicherheit & Härtung

- **HTTPS** für API und Portal einrichten (siehe 7.1) – für den Produktivbetrieb **dringend empfohlen**.
- **PostgreSQL** härten: nur benötigte Netze, starke Passwörter, App-Rolle mit minimalen Rechten.
- **Dienstkonten** mit minimalen Rechten betreiben.
- **AD/LDAP-Anbindung** ist read-only.
- Details und Kontrollrahmen: ISMS-Modul (ID-28).

7.1 HTTPS einrichten

> **Sicherheitshinweis:** Ohne HTTPS laufen Anmeldung und sämtliche Daten **unverschlüsselt** über das Netz. Für den Produktivbetrieb ist HTTPS für API **und** Portal dringend zu verwenden.

Ordivis Platform kennt **drei Betriebsarten** der Transportverschlüsselung (Abschnitt `Hosting`, gleich strukturiert für API und Portal). Sie sind bei der Installation wählbar und lassen sich jederzeit als **SuperAdmin** unter *Einstellungen*

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

→ *Plattform* → *HTTPS / Transportsicherheit* ändern. Weil Kestrel die Endpunkte nur beim Start liest, wird eine Änderung **erst nach einem Neustart** des Dienstes `InfraDesk.API` wirksam (der SuperAdmin-Dialog weist darauf hin).

Der SuperAdmin-Wunschzustand wird in `C:\ProgramData\InfraDesk\config\hosting.json` abgelegt. Diese Datei liegt **außerhalb** des Installationsverzeichnisses und **überlebt Produkt-Updates** (im Gegensatz zur mitgelieferten `appsettings.json`).

Betriebsart	Wann?	Was passiert
Reverse-Proxy (Standard)	TLS terminiert ein vorgelagerter Proxy (IIS/nginx/Traefik)	Der Dienst bindet nur HTTP; ein direkter, unverschlüsselter Zugriff auf den Port ist ein Risiko.
Eigenes Zertifikat	PEM-Text einfügen, vorhandene PFX-Datei oder Windows-Zertifikat	Kestrel bindet HTTP **und** HTTPS mit diesem Zertifikat.
Let's Encrypt (ACME)	öffentlich erreichbarer Server mit Domäne	Zertifikat wird automatisch bezogen und erneuert (HTTP-01-Challenge über Port 80).

a) Eigenes Zertifikat – als Text einfügen (empfohlen):

Zertifikate aus einer CA-Oberfläche (OPNsense, XCA, interne bzw. Behörden-PKI) liegen fast immer als **PEM** vor: eine Zertifikatsdatei (`*.crt`), eine Schlüsseldatei (`*.key`) und meist eine Kettendatei. Diese drei Inhalte lassen sich direkt einfügen; die Umwandlung übernimmt der Server.

1. SuperAdmin-Dialog → *HTTPS / Transportsicherheit* → Betriebsart *Eigenes Zertifikat* → Abschnitt **„Zertifikat als Text einfügen (PEM)"**.
2. Inhalt der Dateien in die drei Felder kopieren. Ist der private Schlüssel verschlüsselt (`-----BEGIN ENCRYPTED PRIVATE KEY-----`), zusätzlich sein Kennwort angeben.
3. **„Prüfen und übernehmen"**. Es wird sofort gemeldet, ob Zertifikat und Schlüssel zusammenpassen, bis wann es gilt und **für welche Namen** – bei einem Fehler steht der Grund direkt dort.
4. Domäne und HTTPS-Port kontrollieren, speichern → Dienst `InfraDesk.API` neu starten.

Die Ausstellerkette ist optional, aber **empfohlen**: Der Server schickt sie beim Verbindungsaufbau mit. Ohne sie funktioniert es häufig, doch manche Klienten – und nahezu jedes Mobilgerät – lehnen die Verbindung ab.

Was mit dem Schlüssel geschieht: Aus den eingefügten Inhalten erzeugt der Server eine PFX-Datei unter `C:\ProgramData\InfraDesk\config\certs\`, die **nur** SYSTEM und Administratoren lesen dürfen. Ihr Kennwort wird zufällig vergeben und verschlüsselt hinterlegt – Sie müssen es weder kennen noch eingeben. Der private Schlüssel liegt damit an keiner Stelle im Klartext.

Der Name im Zertifikat entscheidet: Maßgeblich ist der *alternative Name* (SAN), **nicht** der Common Name – kein heutiger Browser wertet den CN noch aus. Ein brauchbares Zertifikat trägt den Hostnamen, unter dem der Server aufgerufen wird; sinnvollerweise zusätzlich seine IP-Adresse, damit der Zugriff auch bei DNS-Störungen ohne Warnung funktioniert. Fehlt der SAN, sagt die Übernahme das ausdrücklich.

a2) Eigenes Zertifikat – vorhandene PFX-Datei:

1. PFX (mit privatem Schlüssel) auf den Server legen, z. B. `C:\ProgramData\InfraDesk\certs\server.pfx`.
2. SuperAdmin-Dialog: Betriebsart *Eigenes Zertifikat*, Pfad + Kennwort eintragen (alternativ ein Thumbprint aus `LocalMachine\My`), Domäne und HTTPS-Port (Standard 5001) setzen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3. Speichern → Dienst `InfraDesk.API` neu starten.

b) Let's Encrypt / ACME:

1. Voraussetzung: Der Server ist unter der Domäne **aus dem Internet** erreichbar und **Port 80** ist offen (Let's Encrypt prüft die Domäne über `http://{Domäne}/.well-known/acme-challenge/…`).
2. SuperAdmin-Dialog: Betriebsart *Let's Encrypt*, Domäne + Kontakt-E-Mail eintragen, Nutzungsbedingungen akzeptieren. **Test-Umgebung (Staging)** zunächst eingeschaltet lassen – sie umgeht die strengen Mengen-Grenzen; das Zertifikat ist dann aber **nicht** öffentlich vertrauenswürdig.
3. Speichern → Dienst neu starten. Das Zertifikat wird beim Start im Hintergrund bezogen; die Erneuerung läuft automatisch (Prüfung alle 12 h, Erneuerung ab 30 Tagen vor Ablauf) **ohne** weiteren Neustart.
4. Nach erfolgreichem Test **Staging ausschalten** und erneut starten, um ein produktives Zertifikat zu beziehen.

c) Reverse-Proxy (Standard): Der Dienst bleibt auf HTTP; die Verschlüsselung übernimmt der vorgelagerte Proxy. Den HTTP-Port dann **nicht** direkt aus dem Netz erreichbar machen (nur der Proxy spricht ihn an).

d) Self-Service-Portal – dasselbe Zertifikat, eigene Ports:

Das Portal ist ein eigener Windows-Dienst und braucht deshalb **eigene Ports** – zwei Dienste können denselben Port nicht binden. Betriebsart, Domäne und **Zertifikat teilt es sich mit der API**; ein zweites Zertifikat ist nicht nötig.

1. SuperAdmin-Dialog → Abschnitt „**Ports des Self-Service-Portals**": HTTPS-Port auf **443** setzen (Vorgabe ist `0` = kein eigenes HTTPS fürs Portal).
2. Speichern → Dienst `InfraDesk.Portal` neu starten.
3. **Firewall-Freigabe für Port 443 anlegen**, falls das Portal-HTTPS erst nachträglich eingeschaltet wurde – der Installer legt sie nur an, wenn beim Installieren schon ein HTTPS-Modus gewählt war:

```
powershell New-NetFirewallRule -DisplayName "Ordivis Portal (HTTPS)" -Direction Inbound ` -Protocol TCP -LocalPort 443 -Action Allow -Profile Any
```

> **Warum 443 empfohlen ist:** Das Portal ist die Adresse, die Sie an die Belegschaft weitergeben. Auf dem Standard-Port ist es ohne Portangabe erreichbar (`https://helpdesk.ihre-domäne.de`). Die API bleibt auf 5001 – sie wird nicht von Hand aufgerufen.
>
> **Links in E-Mails** (Kennwort zurücksetzen, Termin zusagen, Bewertung) werden aus Domäne und Portal-Port **abgeleitet**, sobald HTTPS mit einer Domäne eingerichtet ist. Steht das Portal auf 443, enthalten die Mails saubere Adressen ohne Portangabe.
>
> **Let's Encrypt und das Portal:** Das Portal bezieht **kein eigenes** Zertifikat – die Domänenprüfung verlangt Port 80, und den kann nur einer der beiden Dienste halten. Es übernimmt das von der API bezogene Zertifikat automatisch, auch nach jeder Erneuerung.

7.2 Client-Verbindung (API-Adresse)

Der Windows-Client muss wissen, unter welcher Adresse die API erreichbar ist – in der Regel laufen API und Datenbank auf **einem** Server. Diese Adresse wird bei der **Client-Installation** abgefragt und ist danach jederzeit änderbar:

- **Auf der Anmeldeseite** über *Verbindungseinstellungen* (auch **vor** der Anmeldung – eine falsche Adresse ließe sich sonst nicht korrigieren).
- **In der Anwendung** als SuperAdmin unter *Einstellungen → Plattform → API-Verbindung*.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Die Adresse (z. B. `https://server.firma.local:5001`) wird **lokal je Arbeitsplatz** gespeichert (nicht am Server, da sie schon vor der Anmeldung gebraucht wird) und beim nächsten Anmelden wirksam. Mit `https://` verbindet sich der Client verschlüsselt; die Schaltfläche *Verbindung testen* prüft die Erreichbarkeit.

Zertifikatfehler ignorieren (selbstsigniertes / internes Zertifikat): Nutzt der Server ein selbstsigniertes oder ein von einer internen CA ausgestelltes Zertifikat, das der Arbeitsplatz **nicht als vertrauenswürdig** kennt, würde der Client die HTTPS-Verbindung normalerweise ablehnen. Für kleine Installationen ohne eigene PKI gibt es im Verbindungsdialog die Option **„Zertifikatfehler bei HTTPS ignorieren"**. Damit ist die Verbindung **verschlüsselt, aber nicht auf Echtheit geprüft** – besser als gar kein HTTPS, aber ein bewusster Kompromiss (theoretisch Man-in-the-Middle möglich). Der Client zeigt in diesem Fall dauerhaft den Hinweis *„Zertifikatprüfung deaktiviert"* an.

> **Richtig gemacht** (empfohlen, statt Fehler dauerhaft zu ignorieren): Ein **vertrauenswürdiges** Zertifikat verwenden – entweder von einer öffentlichen CA / Let's Encrypt (siehe 7.1) **oder** das Zertifikat der **internen CA** einmalig auf den Arbeitsplätzen als vertrauenswürdig hinterlegen (Zertifikat der Stammzertifizierungsstelle in den Windows-Speicher *Vertrauenswürdige Stammzertifizierungsstellen* importieren, z. B. per Gruppenrichtlinie). Danach die Option wieder abschalten. Für das **Portal→API** gibt es dieselbe Möglichkeit über die Portal-Einstellung `ApiIgnoreCertificateErrors: true`.

7.3 TLS-Version

Ordivis Platform gibt die TLS-Version **nicht fest vor**, sondern überlässt sie dem Betriebssystem (`SslProtocols.None` – von Microsoft empfohlen, damit das OS die beste verfügbare Version wählt und über Windows-Updates gepflegt wird). In der Praxis heißt das: **TLS 1.3 wird automatisch genutzt, sobald das Server-Betriebssystem es unterstützt** (Windows 11 / Windows Server 2022 und neuer), sonst **TLS 1.2** als Rückfall. Ältere Versionen (TLS 1.0/1.1) sind auf aktuellen Windows-Systemen bereits per Schannel-Standard deaktiviert. Eine strengere Mindestversion lässt sich bei Bedarf über die **Schannel-Härtung des Servers** (Registry/Gruppenrichtlinie) erzwingen.

7.4 Zugänge für Geräte und Programme (Token)

Nicht jeder Zugriff auf Ordivis Platform kommt von einem Menschen an einer Tastatur. Ein Scan-Werkzeug im Nebenstandort und ein Infobildschirm im Empfangsbereich brauchen beide einen dauerhaften Zugang – aber für keinen von beiden darf ein Kennwort auf einem fremden Rechner liegen. Dafür gibt es **Token**: langlebige Anmelde-Merkmale, die einzeln benannt und einzeln widerrufen werden können.

Alle Token verwalten Sie an einer Stelle: **Einstellungen → Plattform → Integrations-Token**. Die Seite trennt sie in zwei Abschnitte, weil sie unterschiedlich funktionieren:

	Externe Programme	Anzeigegeräte (Wallboard)
Wofür	Werkzeuge, die Daten per Schnittstelle einspeisen	Bildschirme, die unbeaufsichtigt Kennzahlen zeigen
Wer ist der Zugriff	der Token selbst – kein Benutzerkonto	ein **Benutzerkonto**, an das der Token gebunden ist
Rechte	fest auf das Einspeisen von Scan-Ergebnissen begrenzt	die Rechte des gebundenen Kontos
Laufzeit	freiwillig	**Pflicht** (1 bis 730 Tage)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Gemeinsam ist beiden: Der **rohe Token wird nur einmal bei der Ausstellung angezeigt**; gespeichert wird nur sein Hash. Geht er verloren, stellen Sie einen neuen aus und widerrufen den alten.

> **Berechtigung:** Der Abschnitt „Externe Programme" verlangt `platform.integrationtoken.manage`, der Abschnitt „Anzeigegeräte" `system.config.manage`. Wer nur eines der beiden Rechte hält, sieht auch nur den zugehörigen Abschnitt.

7.4.1 EXTERNE PROGRAMME (SCAN-INGEST)

Externe Administrations-Werkzeuge – etwa das kostenlose **Ordivis Toolbox** – können Netzwerkscan-Ergebnisse über eine Schnittstelle in die **IPAM** einspeisen. Das ist besonders für **entfernte Subnetze** nützlich, die der Ordivis-Server selbst nicht erreicht (andere Standorte, isolierte Segmente): Das Werkzeug läuft auf einem Admin-Arbeitsplatz mit Sicht auf das Netz und übermittelt die Ergebnisse an Ordivis Platform. Für diese Anbindung wird **kein Benutzer-Login** verwendet, sondern ein **Integrations-Token** (Maschinen-Zugang).

Token ausstellen (nur SuperAdmin):

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Integrations-Token* · 🔑 `tenant.manage`

🔒 **Pflichtreihenfolge** — der rohe Token wird **einmal** angezeigt. Halten Sie das Zielsystem bereit, bevor Sie ihn ausstellen.

1. Im Client *Einstellungen* ▸ *Integrations-Token* öffnen, Abschnitt **Externe Programme**.
2. **Token für Programm** wählen, einen sprechenden **Namen** vergeben (z. B. „Ordivis Toolbox – Standort Nord") und optional ein **Ablaufdatum** setzen.
3. Der **rohe Token wird nur einmalig angezeigt** – kopieren und im externen Werkzeug hinterlegen. Danach ist er nicht mehr abrufbar (gespeichert wird nur ein Hash). Bei Verlust einen neuen ausstellen.

Eigenschaften und Grenzen:

- Ein Token gilt für **genau einen Mandanten** und trägt nur die **Einspeise-Rechte** – keinen allgemeinen Lese- oder Schreibzugriff auf andere Daten und keinen Zugang zur Oberfläche.
- Token lassen sich jederzeit **widerrufen**; ein Agent, der einen widerrufenen Token nutzt, wird sofort mit **401** abgewiesen.
- Eingespeiste Daten werden **nicht blind übernommen**: Neue Geräte erscheinen als **unbestätigte** IP-Adressen und neu angelegte Subnetze im Status **„Reserviert"**. Eine Administratorin prüft und bestätigt sie anschließend in der IPAM.
- **Rechte wachsen einem Token nicht nachträglich zu.** Ein vor einer Erweiterung ausgestellter Token erreicht die neuen Endpunkte nicht (Antwort **403**) – dafür muss ein neuer Token ausgestellt werden. Das ist beabsichtigt: Was ein Maschinen-Zugang darf, wird bei seiner Ausstellung entschieden.

Was ein Werkzeug einspeisen kann:

Datenquelle	Endpunkt	Landet als
Netzwerkscan	`POST /api/v1/ipam/ingest/scan`	unbestätigte IP-Adressen, Subnetz „Reserviert"
DHCP-Bereiche	`POST /api/v1/ipam/ingest/dhcp`	Subnetze + Reservierungen (Vorschau möglich)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Datenquelle	Endpunkt	Landet als
DNS-/PTR-Abgleich	`POST /api/v1/ipam/ingest/dns`	Namen an bekannten Adressen; Widersprüche werden **gemeldet**, nicht bereinigt
Geräte, AD-Computer, Software	`POST /api/v1/discovery/ingest/devices`	Funde im **Client-Inventar** (Prüfstand)
Zertifikate	`POST /api/v1/discovery/ingest/certificates`	CI der Klasse „Zertifikat" im Zustand **Staging**, nicht verifiziert
Schwachstellen (CVE/WID)	`POST /api/v1/security/ingest/findings`	Risiken in der Analyse „Schwachstellen (Scan)" des Informationsverbunds

Zwei Dinge kommen dabei **nie** an: **LAPS-Kennwörter** und **BitLocker-Wiederherstellungsschlüssel**. Die Werkzeuge übertragen sie nicht, auch wenn sie sie im Verzeichnis sehen könnten.

Erreichbarkeit: Alle Ingest-Endpunkte laufen über dasselbe HTTPS wie die übrige API (siehe **7.1 HTTPS einrichten**). Verbindungstests: `GET /api/v1/ipam/ingest/health` und `GET /api/v1/discovery/ingest/health`; die Werkzeuge bieten dafür üblicherweise eine Schaltfläche „Verbindung testen".

> **Sicherheitshinweis:** Behandeln Sie Integrations-Token wie Passwörter. Stellen Sie pro Werkzeug/Standort einen eigenen Token aus (bessere Nachvollziehbarkeit und gezielter Widerruf) und vergeben Sie bei Bedarf ein Ablaufdatum.

7.4.2 ANZEIGEGERÄTE (WALLBOARD)

Ein **Wallboard** ist ein Bildschirm, der unbeaufsichtigt läuft und im Wechsel Kennzahlen zeigt – im Service-Desk-Büro, am Empfang, im Leitstand. Das Gerät soll nach einem Stromausfall von allein wieder hochkommen, ohne dass jemand ein Kennwort eintippt. Genau dafür ist dieser Token da; die Bedienung des Bildschirms selbst beschreibt das Kapitel **Reporting & Dashboards**.

Token ausstellen:

> 💻 **Nur im Client:** *Fußbereich* ▸ *Einstellungen* ▸ *Integrations-Token* · 🔑 `tenant.manage`

🔒 **Pflichtreihenfolge** — auch hier wird der rohe Token nur **einmal** angezeigt.

1. *Einstellungen* ▸ *Integrations-Token* öffnen, Abschnitt **Anzeigegeräte (Wallboard)**.
2. **Token für Anzeigegerät** wählen und ausfüllen: - **Name des Anzeigegeräts** – wo der Schirm hängt (z. B. „Empfang EG"). Danach wird er einmal gezielt widerrufen, wenn das Gerät ausgetauscht wird. - **Konto, unter dessen Rechten der Schirm liest** – nur aktive Konten stehen zur Auswahl. - **Laufzeit in Tagen** – Vorgabe 365, höchstens 730.
3. Der **rohe Token wird nur einmalig angezeigt** – kopieren und auf dem Anzeigegerät in der Wallboard-App unter **Einrichtung → Wallboard-Token** eintragen.

Was der Schirm sehen darf, entscheidet das Konto. Der Token trägt keine eigenen Rechte; jede Anfrage wird mit den Rechten des gebundenen Kontos beantwortet.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Sie können einen Token nur für ein Konto ausstellen, dessen Rechte Sie selbst vollständig haben. Seit Version 2026.07.28.1912 gilt hier dieselbe Schranke wie bei der Rollenvergabe. Der Grund: Ein Wallboard-Token wird **anonym und ohne zweiten Faktor** gegen ein Zugriffstoken mit den Rechten des gebundenen Kontos eingetauscht. Ohne diese Schranke hätte, wer Anzeigegeräte verwalten darf, sich über ein vorgebliches „Wallboard" die Rechte eines höher privilegierten Kontos verschaffen können – ohne Kennwort und ohne dass eine Rollenzuweisung protokolliert worden wäre. Steht ein Konto nicht zur Auswahl, fehlen Ihnen selbst Rechte, die dieses Konto hat.

Empfehlung: Legen Sie für Wallboards ein **eigenes Konto mit ausschließlich lesenden Rechten** an und verwenden Sie **nicht** ein persönliches Konto. Ein Bildschirm im Flur zeigt allen Vorbeigehenden, was dieses Konto sehen darf – und ein gestohlener Token gäbe genau diese Sicht in fremde Hände.

Was beim Widerruf passiert: Der Schirm zeigt beim nächsten Erneuern der Sitzung einen Hinweis statt Zahlen. Ein laufendes Wallboard fällt also nicht sofort aus, aber innerhalb kurzer Zeit.

Ablage auf dem Gerät: Die Wallboard-App speichert den Token mit **Windows-DPAPI**, gebunden an das Windows-Konto, unter dem sie eingerichtet wurde. Läuft das Wallboard später unter einem anderen Windows-Konto, muss der Token dort neu eingetragen werden – die alte Datei lässt sich nicht entschlüsseln.

8 Skripte & Encoding

Achtung: PowerShell-Skripte (`.ps1`) mit Sonderzeichen **müssen UTF-8 mit BOM** sein, sonst Parse-Fehler unter PowerShell 5.1 auf dem Server.

9 Fehlerbehebung (Auszug)

Symptom	Ursache / Abhilfe
Dienst startet nach Update nicht	SCM-Timeout → erneut `Start-Service`; ggf. Prozess elevated killen
404 statt 401 nach Update	DLLs gesperrt (Dienst nicht rechtzeitig gestoppt) → `taskkill`, neu deployen
API-Fehler 42501 / DbUpdate	Tabellen gehören `postgres` → `ALTER TABLE ... OWNER TO infradesk_app`
Client-Absturz nach Splash	fehlende PRI (Publish statt Build) oder fehlende StaticResource → `%LOCALAPPDATA%\InfraDesk\Logs\client-error.log` (das Startprotokoll aus der Zeit, bevor das reguläre Protokoll läuft)
E-Mail-Versand scheitert	Root-CA fehlt / VPN nötig / `Notifications:Smtp` prüfen
Discovery entschlüsselt nicht	Vault-Schlüssel passt nicht zu Geheimnissen → Zugangsdaten neu hinterlegen
500 bei Zeitwerten	DateTimeOffset muss vor dem Speichern `.ToUniversalTime()` sein (timestamptz)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Symptom	Ursache / Abhilfe
Update meldet „kein gültig signiertes Manifest"	**Nicht der Meldung glauben, sondern das Update-Protokoll lesen** (`update-logs\` , siehe 10.4). Steht dort „Staging": Das Manifest ist in Ordnung – es fehlen Dateien auf dem Update-Server. Welche, sagt `api-JJJJMMTT.log` .
Update meldet „Es läuft bereits ein Update-Vorgang"	Es läuft immer nur **ein** Update gleichzeitig. Die Meldung nennt seit wann – und, falls der Vorgang abgebrochen ist, wann die Sperre von selbst freigibt (30 Minuten nach dem Start) sowie den Pfad der Statusdatei. Ein zweiter Anlauf davor wird abgewiesen; das ist kein Fehler.
LDAP-Sync meldet nur „fehlgeschlagen"	Ursache steht als innere Ausnahme im `api-JJJJMMTT.log` . Binäre Verzeichnisattribute werden beim Abgleich übersprungen (siehe Kapitel 41, Abschnitt 8.1) – schlägt der Sync **jedes Mal** fehl, ist die Ursache in aller Regel die Verbindung oder das Konto, nicht ein einzelner Datensatz.

10 Logs

Jede Komponente schreibt ihre eigene Logdatei – und zwar lokal auf dem Rechner, auf dem sie läuft. Der Dateiname nennt die Quelle, man sieht also auf einen Blick, wer geschrieben hat:

Datei	Anwendung	Wo	Dienst
`api-JJJJMMTT.log`	API / Anwendungsserver	**Server**, `C:\ProgramData\InfraDesk\Logs`	`InfraDesk.API`
`portal-JJJJMMTT.log`	Self-Service-Portal	**Server**, ebd.	`InfraDesk.Portal`
`worker-JJJJMMTT.log`	Discovery-Collector	**Rechner im Zielnetz**, ebd.	`InfraDesk.Discovery.Collector`
`updateagent-JJJJMMTT.log`	Update-Agent (Mehr-Server-Betrieb)	**Server**, ebd.	`InfraDesk.UpdateAgent`
`updater-JJJJMMTT.log`	Aktualisierung (Arbeiter + Fortschrittsfenster)	**Server / Arbeitsplatz**, ebd.	–
`clientupdate-JJJJMMTT.log`	Client-Update-Dienst	**Arbeitsplatz**, ebd.	`InfraDesk.ClientUpdate`
`client-JJJJMMTT.log`	Windows-Client	**Arbeitsplatz**, `%LOCALAPPDATA%\InfraDesk\Logs`	–
`wallboard-JJJJMMTT.log`	Wallboard	**Anzeigegerät**, ebd.	–
`mobile-JJJJMMTT.log`	Mobile App (Android)	**Mobilgerät**, im privaten Datenbereich der App	–

Die Dateien **rollieren automatisch**: im eingestellten Takt ein neuer Stand, zusätzlich bei Überschreiten der Größe. Alte Stände über die eingestellte Anzahl hinaus werden selbsttätig gelöscht – die Logs können also nicht unbemerkt die Platte füllen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Warum der Client ins Benutzerprofil schreibt: Er läuft im Kontext des angemeldeten Benutzers und hat unter `ProgramData` je nach Installation kein Schreibrecht. Außerdem würden sich auf einem Terminalserver sonst alle Benutzer in dieselbe Datei schreiben. Dasselbe gilt für Wallboard und App.

10.1 Einstellen im Client (empfohlener Weg)

Der Weg über die Oberfläche ist der **maßgebliche**: **Einstellungen → Protokolle (Logs)**. Was dort gespeichert wird, liegt in `%ProgramData%\InfraDesk\config\operations.json`, überlebt ein Produkt-Update und gilt für API, Portal und Collector gemeinsam. Der Windows-Client holt sich seine Stufe beim Anmelden von dort ab. Die Seite verlangt die Plattform-Berechtigung `platform.hosting.manage` (SuperAdmin).

Einstellbar sind:

Bereich der Seite	Was sich einstellen lässt
Protokollierung	Hauptschalter über alle Anwendungen.
Speicherort	Zielordner der Server-Dienste. Empfehlung: nicht auf `C:` , sondern auf ein Datenlaufwerk.
Ausführlichkeit	Mindeststufe: `Debug` · `Information` · `Warnung` · `Fehler` .
Rotation & Aufbewahrung	Takt (stündlich / täglich / monatlich / jährlich / nie), Anzahl aufbewahrter Dateistände, Größe für einen zusätzlichen Umbruch, Format (Text oder JSON/CLEF).
Anwendungen	Ein/Aus je Anwendung – und optional eine eigene Stufe für genau eine davon.
Was genau protokolliert wird	Mindeststufe je Bereich (Technik, Anwendungen, Fachmodule) – etwa „Datenbank (SQL)" auf `Debug` , während alles andere ruhig bleibt.

Was sofort wirkt und was nicht. Stufen und Schalter greifen **ohne Dienst-Neustart** – genau darauf kommt es an, wenn gerade jemand am Telefon ist: umstellen, den Vorgang wiederholen, zurückstellen. Speicherort, Rotationstakt, Aufbewahrung und Format legt die Anwendung beim Start fest; sie wirken erst beim nächsten Start. Die Seite sagt nach dem Speichern, welcher der beiden Fälle eingetreten ist.

Fehler kommen immer durch. Auch bei Stufe „Fehler" wird jeder Fehler geschrieben – die Stufe legt nur fest, wie viel *zusätzlich* protokolliert wird. Für eine Fehlersuche vorübergehend auf `Debug` stellen, danach zurückstellen: Auf `Debug` wachsen die Dateien schnell.

„Debug" macht die Laufzeitumgebung absichtlich nicht gesprächig. Die technischen Bereiche (`.NET` , Datenbank, HTTP) bleiben auch dann auf „Warnung" – sonst stünde in jeder zweiten Zeile eine Datenbankabfrage und die eigentliche Spur ginge unter. Wer die SQL-Anweisungen sehen will, stellt den Bereich **„Datenbank (SQL)"** gezielt auf `Debug` .

10.2 Einstellen in der Datei (`appsettings.json`)

Für unbeaufsichtigte Installationen und für Rechner ohne Server (eigener Collector, Arbeitsplatz) steht derselbe Abschnitt in der `appsettings.json` der jeweiligen Anwendung:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
"Logging": {
  "File": {
    "Enabled": true,
    "Directory": "C:\\ProgramData\\InfraDesk\\Logs",
    "Level": "Information",
    "RollingInterval": "Day",
    "RetainedDays": 31,
    "FileSizeLimitMb": 50,
    "Format": "Text",
    "Components": { "wallboard": { "Enabled": false } },
    "Areas": { "database": "Debug" }
  }
}
```

Einstellung	Bedeutung
`Enabled`	Datei-Protokoll an (`true`) oder aus (`false`). **Vorgabe: an.**
`Directory`	Zielordner. Muss für das Dienstkonto beschreibbar sein. Leer = Vorgabe der Anwendung.
`Level`	`Debug` (sehr ausführlich, für die Fehlersuche) · `Information` (Vorgabe) · `Warning` · `Error`
`RollingInterval`	Wann eine neue Datei beginnt: `Hour` · `Day` (Vorgabe) · `Month` · `Year` · `Infinite` (nie).
`RetainedDays`	Wie viele **Dateistände** aufbewahrt werden (Vorgabe 31). Nicht Tage: Bei Größenumbruch entstehen mehrere Stände pro Tag.
`FileSizeLimitMb`	Ab dieser Größe wird auch innerhalb des Takts eine neue Datei begonnen.
`Format`	`Text` (lesbar, Vorgabe) oder `Json` – eine JSON-Zeile je Ereignis im Format **CLEF**.
`Components`	Ein/Aus (und optional eine eigene Stufe) je Anwendung. Schlüssel: `api` , `portal` , `worker` , `client` , `wallboard` , `clientupdate` , `updater` , `updateagent` , `mobile` .
`Areas`	Mindeststufe je Bereich, z. B. `database` , `http` , `ticketing` , `identity` .

> **Rangfolge:** Was in der Oberfläche gespeichert wurde (`operations.json`), liegt **über** der `appsettings.json` . Die Datei ist also die Auslieferungsvorgabe, nicht das letzte Wort.

Eine Zeile sieht so aus – Zeitpunkt, Stufe, **Quelle**, Meldung und der Zusatzkontext:

```
2026-08-08 09:12:03.118 +02:00 [INF] [InfraDesk.Modules.Ticketing.TicketService] Vorgang T-1042 ges
{"CorrelationId":"7f3c…","TenantId":"…","User":"m.schulz","Component":"api","AppVersion":"2026.08.0
```

Die **Korrelations-ID** ist dabei der Faden durch die Anlage: Client, Portal und App schicken sie mit, die Schnittstelle übernimmt sie und gibt sie zurück. Dieselbe Zeichenfolge steht danach in `client-JJJJMMTT.log` **und** in `api-JJJJMMTT.log` – eine Meldung vom Arbeitsplatz lässt sich damit ohne Raten der passenden Server-Anfrage zuordnen.

10.2a Format `Json` (CLEF) für SIEM und Log-Agenten

Stellen Sie das Format auf `Json` , schreibt jede Anwendung **eine JSON-Zeile je Ereignis**. Das lesen Fluent Bit, Filebeat, Vector, Seq, Elastic und Loki ohne eigene Zerlegungsregel ein, und jedes Zusatzfeld (Korrelations-ID,

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Mandant, Benutzer, Version, Rechner) bleibt ein eigenes Feld statt in einer Textzeile zu verschwinden. Für das Lesen von Hand ist `Text` besser geeignet.

10.3 Zentrale Sammlung (mehrere Server, SIEM)

Verteilt Ordivis Platform sich über mehrere Server (API und Portal getrennt), stellt sich die Frage nach einer zentralen Auswertung. Dafür gilt ein Grundsatz:

> **Schreiben Sie die Logs nicht direkt auf eine Netzwerkfreigabe.** Das ist naheliegend und trotzdem falsch: Ein Log auf einem SMB-Share fällt **genau dann aus, wenn man es braucht** – bei einer Netzwerkstörung. Es kann die Anwendung blockieren (ein Schreibzugriff über SMB kann sekundenlang hängen), es erfordert Schreibrechte des Dienstkontos auf einer Freigabe, und mehrere Server kollidieren in derselben Datei.

Der übliche Weg trennt **Schreiben** und **Einsammeln** in zwei Schritte. Das Einsammeln darf ausfallen, ohne dass Ordivis Platform davon berührt wird:

Verfahren	Wann sinnvoll
Windows Event Forwarding (WEF)	Bordmittel, kein Zusatzserver. Die Dienste schreiben ohnehin ins Ereignisprotokoll; WEF sammelt es auf einem Collector ein. Der einfachste Einstieg.
Log-Agent (Fluent Bit, Filebeat, Winlogbeat, Promtail …)	Liest die Dateien aus `Logs\` und liefert sie an Elastic, Loki, Graylog, Seq oder Splunk. Der Standardweg, wenn eine Log-Plattform vorhanden ist.
Syslog / SIEM	In Behörden verbreitet und für ein ISMS oft ohnehin gefordert. Ein Syslog-Agent liest dieselben Dateien.
Geplante Kopie auf eine Freigabe	Für kleine Installationen ausreichend: ein Task, der die Logs nachts wegsichert. **Kopieren statt hineinschreiben** – die Anwendung merkt nichts davon.

Ordivis Platform schreibt bewusst **nur lokal** und macht keine Vorgaben darüber, womit Sie einsammeln.

10.4 Weitere Protokolle

- **Update-Läufe:** `C:\ProgramData\InfraDesk\update-logs\update_<version>_<zeit>.log` – jeder Schritt eines Update-Laufs mit Zeitstempel, **eine Datei je Lauf**. **Bei jedem Update-Problem als Erstes hier hineinsehen**: Die Meldung im Client ist knapp, dieses Protokoll nennt den tatsächlichen Abbruchgrund. Dieselben Zeilen stehen zusätzlich in `updater-JJJJMMTT.log` und unterliegen dort Ihrer Einstellung.
- **Ereignisprotokoll (Windows):** Die Dienste schreiben zusätzlich dorthin – nützlich, wenn ein Dienst gar nicht erst startet und deshalb keine Datei anlegen konnte.
- **Startprotokolle:** `client-error.log` (Arbeitsplatz) und `bootstrapper_<zeit>.log` (Update) halten fest, was **vor** dem Hochfahren des regulären Protokolls passiert ist. Sie unterliegen daher weder Stufe noch Rotation – dafür gibt es sie auch dann, wenn eine Anwendung gar nicht erst so weit kommt.
- **Audit-Log:** revisionssichere Fachaktionen, einsehbar im Client (ID-27). Das ist ein Fach-, kein Technikprotokoll.

10.5 Support-Paket – ein Klick statt zehn Rückfragen

Wenn Sie beim Hersteller-Support anrufen, braucht dieser als Erstes den **Zustand Ihrer Anlage**. Statt ihn über mehrere Rückfragen zusammenzutragen, erzeugen Sie ihn in einem Schritt:

Einstellungen → Protokolle (Logs) → „Support-Paket erstellen"

Das Paket enthält:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Teil	Inhalt
`00-STECKBRIEF.txt`	Lizenznehmer und Produkt, alle Mandanten mit Status, installierte Versionen je Anwendung, Betriebssystem, Laufzeitumgebung, Rechnername, Startzeit des Dienstes, Zeitzone – und Ihre Schilderung des Vorfalls.
`01-INHALT.txt`	Was aufgenommen wurde **und was nicht, mit Grund**.
`logs/`	Die Protokolldateien aller Anwendungen der gewählten Anzahl Tage (Vorgabe 3).
`update-logs/`	Die Protokolle der Update-Läufe samt Zustand des letzten Laufs.
`config/`	Die Konfigurationsdateien – **Kennwörter, Token und Schlüssel geschwärzt**.
`eventlog.txt`	Die Ordivis-Einträge des Windows-Anwendungsprotokolls.

Was NICHT enthalten ist: keine Fachdaten (keine Vorgänge, Objekte oder Benutzerkonten), keine Datenbankinhalte und keine Geheimnisse. Das Paket beschreibt den *Betriebszustand*, nicht den Inhalt Ihrer Anlage – es lässt sich also ohne lange Abwägung verschicken.

Das Archiv ist ein **7z-Archiv mit AES-256**, einschließlich der Dateinamen: Ohne Kennwort lässt sich nicht einmal auflisten, was darin liegt.

Das Kennwort steht bewusst NICHT in der Mail. Ein verschlüsseltes Archiv, dessen Kennwort danebensteht, ist gegenüber jedem, der die Mail sieht, nicht verschlüsselt. Sie bekommen es im Dialog angezeigt (fünf Vierergruppen, ohne verwechselbare Zeichen wie `0` / `O` – es lässt sich vorlesen) und geben es dem Support auf dem Weg weiter, auf dem Sie ohnehin gerade sind: im laufenden Telefonat.

Versendet wird an **support@ordivis.eu**, über das E-Mail-Postfach Ihrer Organisation – der Support kann also antworten. Zwei Fälle, in denen nicht versendet wird und das Paket stattdessen zum **Herunterladen** bereitsteht:

- Sie haben den Versand abgewählt.
- Das Paket ist größer als die Grenze für E-Mail-Anhänge (10 MB). Dann sagt der Dialog das ausdrücklich – wählen Sie weniger Tage oder übermitteln Sie die Datei auf anderem Weg.

Jede Erstellung wird im Windows-Sicherheitsprotokoll vermerkt (wer, wann, welche Datei, an wen) – das Kennwort selbstverständlich nicht.

Abgrenzung zum Fehlerbericht. Das Käfer-Symbol in der Statusleiste (Kapitel 11) meldet *einen* Vorfall aus Sicht eines Anwenders: Beschreibung, Bildschirmfoto, das Tagesprotokoll seines Arbeitsplatzes. Das Support-Paket ist das Werkzeug des Administrators und beschreibt die ganze Anlage. Beides setzt eine Handlung voraus; es geht nichts im Hintergrund hinaus.

Verwandte Themen

- **ID-02 – Systemvoraussetzungen & Architektur** – Voraussetzungen und Architektur vorab
- **ID-51 – Datenbank-Leistung & Dimensionierung** – Dimensionierung und Datenbank-Leistung
- **ID-40 – Administrationshandbuch** – Konfiguration in der Anwendung
- **ID-30 – Modul Notfall- & Wiederanlaufplanung (DR)** – Wiederanlauf nach einem Ausfall

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-28 – Modul Informationssicherheit (ISMS)** – Härtung und Nachweise

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL VI · BETRIEB

Datenbank-Leistung, Verbindungspools & Dimensionierung

Datenbank-Leistung, Verbindungspools und Dimensionierung je Kundengröße

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-51 Leistung

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Datenbank-Leistung, Verbindungspools & Dimensionierung

Dieses Kapitel richtet sich an **IT-Betrieb / Datenbank-Administratoren**. Es beschreibt jede leistungsrelevante Einstellung im Detail und gibt **empfohlene Werte je Kundengröße** (nach Lizenzstufe) an. Es baut auf den Systemvoraussetzungen (ID-02) und dem Betriebshandbuch (ID-50, Abschnitt 2.3) auf.

INHALT

1 Grundprinzip: zwei Ebenen, ein Zusammenspiel
2 PostgreSQL-Server-Parameter
2.1 Wie Ordivis Platform sie setzt
2.2 Die Parameter im Einzelnen
2.3 Bewusst NICHT gesetzt: statement_timeout
2.4 Werte ändern und zurücknehmen
3 Der Verbindungspool (Database:MaxPoolSizePerTenant)
3.1 Warum es diese Einstellung gibt
3.2 Die Einstellung im Detail
3.3 Ändern
4 Dimensionierung: die Formel
5 Empfohlene Einstellungen je Kundengröße (Lizenz)
5.1 Corporate (Privatwirtschaft)
5.2 Kommunen (Gov)
5.3 MSP (viele Kundenmandanten)
6 Mitgebrachte / externe PostgreSQL-Instanz
7 Prüfen & Überwachen
8 Symptome & Diagnose
Verwandte Themen

1 Grundprinzip: zwei Ebenen, ein Zusammenspiel

Ordivis Platform betreibt jeden Mandanten in einer **eigenen Datenbank** (`infradesk_<slug>`) auf **einem** PostgreSQL-Server (Database-per-Tenant), plus eine zentrale, mandantenfreie **Registry-Datenbank** (`infradesk`). API und Self-Service-Portal laufen in der Standardinstallation auf **derselben Maschine** wie PostgreSQL.

Für die Leistung zählen zwei Ebenen, die zusammenpassen müssen:

- **Server (PostgreSQL):** `max_connections` ist die **harte Obergrenze** aller gleichzeitigen Verbindungen. Jede Verbindung ist bei PostgreSQL ein **eigener Prozess** und kostet Arbeitsspeicher.
- **Client (Npgsql, im API-Dienst):** Npgsql hält je Datenbank einen **eigenen Verbindungspool**. Der Pool bestimmt, **wie schnell** die harte Obergrenze erreicht wird.

Wichtig: Die Zahl der **Mitarbeiter** bestimmt die Verbindungslast **nicht** direkt. Alle Anwender (Agenten im Client, Melder im Portal) laufen über die **API** – diese bündelt sie über ihren Pool. Eine Verbindung wird je Anfrage nur für **Millisekunden** gehalten und danach zurückgegeben. Der eigentliche Engpass skaliert mit der Zahl der **gleichzeitig aktiven Mandanten**, nicht mit der Kopfzahl. Ein Einzelkunde mit 1.000 Mitarbeitern ist **ein** Mandant und unkritisch; kritisch wird es erst bei **vielen** Mandanten (MSP/Kommune).

2 PostgreSQL-Server-Parameter

2.1 Wie Ordivis Platform sie setzt

Bringt der Installer PostgreSQL **selbst** mit, berechnet er die Werte aus der Zielmaschine (Arbeitsspeicher, Prozessorkerne, freier Plattenplatz) und setzt sie per `ALTER SYSTEM` – geschrieben nach `postgresql.auto.conf` . Ihre handgepflegte `postgresql.conf` bleibt unberührt, und jeder Wert lässt sich mit `ALTER SYSTEM RESET <name>` einzeln zurücknehmen (Details in ID-50, Abschnitt 2.3). Eine **mitgebrachte/externe** Instanz wird **nicht** angefasst – dort setzen Sie die Werte selbst (Abschnitt 6).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2.2 Die Parameter im Einzelnen

Parameter	Bedeutung	Herleitung (Installer)	Wirksam ab
`max_connections`	Höchstzahl gleichzeitiger Verbindungen. Jede = ein Prozess.	`8 × GB RAM`, begrenzt auf **100–400**	**Neustart**
`shared_buffers`	Von PostgreSQL selbst verwalteter Cache.	`25 % RAM`, gedeckelt auf **8 GB**	**Neustart**
`effective_cache_size`	*Planer-Hinweis*, wie viel Cache (inkl. OS) verfügbar ist – belegt keinen Speicher.	**50 % RAM** (nicht 75 %, da API und Portal mitlaufen)	Reload
`work_mem`	Speicher **je Sortier-/Hash-Schritt je Verbindung**. Der klassische Weg in den Speicherüberlauf.	`(25 % RAM) ÷ (max_connections × 2)`, **4–16 MB**	Reload
`maintenance_work_mem`	Speicher für Wartung (Index-Aufbau, `VACUUM`). Läuft selten parallel.	`10 % RAM`, gedeckelt auf **2 GB**	Reload
`max_wal_size` / `min_wal_size`	Größe des Write-Ahead-Logs. Größer = weniger Checkpoints, mehr Plattenbedarf.	nach freiem Platz: 1 / 4 / 8 / 16 GB	Reload
`idle_in_transaction_session_timeout`	Trennt Verbindungen, die eine Transaktion offen halten, ohne zu arbeiten.	**60 s**	Reload
`shared_preload_libraries`	Lädt `pg_stat_statements` (Grundlage der Abfrage-Analyse).	ergänzt (überschreibt nicht)	**Neustart**

> **Achtung:** `max_connections` und `shared_buffers` werden erst nach einem **vollständigen Neustart** des Dienstes wirksam – ein `reload` genügt **nicht**. Die übrigen Parameter greifen per `reload` (`SELECT pg_reload_conf();`).

2.3 Bewusst NICHT gesetzt: `statement_timeout`

Ordivis Platform setzt **kein** globales `statement_timeout`. Grund: Die Datenbank-**Migrationen** laufen unter derselben Rolle wie die Anwendung (`infradesk_app`). Ein globales oder rollenweites Zeitlimit würde damit nicht nur lange Abfragen, sondern auch **Index-Neuaufbauten beim Update**, `pg_dump`-Sicherungen und die Größenabfrage des Monitorings abbrechen. Ein Zeitlimit gehört dorthin, wo der Aufrufer bekannt ist – als Npgsql-`CommandTimeout` je Abfrage.

2.4 Werte ändern und zurücknehmen

```
-- Wert setzen (Beispiel), schreibt nach postgresql.auto.conf
ALTER SYSTEM SET max_connections = '300';
-- Einzelnen Wert wieder auf den Standard zurücknehmen
ALTER SYSTEM RESET max_connections;
-- Reload-fähige Parameter ohne Neustart übernehmen
SELECT pg_reload_conf();
```

Für einen Neustart-pflichtigen Parameter anschließend den Dienst neu starten:

```
Restart-Service postgresql-ordivis
```

Das Tuning-Skript lässt sich jederzeit erneut anstoßen (idempotent), zum reinen Anzeigen mit `-DryRun`:

```
powershell -File tune-postgresql.ps1 -PgSuperPassword "<Kennwort>" -DryRun
```

3 Der Verbindungspool (`Database:MaxPoolSizePerTenant`)

3.1 Warum es diese Einstellung gibt

Npgsql führt **je Connection-String einen eigenen Pool**. Bei Database-per-Tenant bedeutet das: **je Mandant ein eigener Pool**. Ohne Grenze poolt Npgsql je Mandant mit dem Standardwert **100** – zwei aktive Mandanten schöpfen ein ungetuntes `max_connections = 100` also bereits rechnerisch aus. Deshalb begrenzt Ordivis Platform den Pool **je Mandant**.

3.2 Die Einstellung im Detail

In `appsettings.json` des API-Dienstes:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
"Database": {
  "MaxPoolSizePerTenant": 50
}
```

Aspekt	Verhalten
Bedeutung	Obergrenze (`Maximum Pool Size`) je Mandanten-Pool **und** für die zentrale Registry-/Haupt-DB.
Standardwert	**50** (gilt auch, wenn der Schlüssel fehlt).
`≤ 0`	Bewusst **unbegrenzt** – fällt auf den Npgsql-Standard 100 zurück.
Wo erzwungen	Zentral beim Auflösen der Mandanten-Verbindung. Greift dadurch **auch für bereits angelegte Mandanten** (deren gespeicherter String die Grenze noch nicht trägt), nicht nur für neue.
Untergrenze	Npgsqls `Minimum Pool Size` bleibt **0** – **leerlaufende** Mandanten-Pools geben ihre Verbindungen nach ~5 min wieder frei.

> **Gut zu wissen:** Weil leerlaufende Pools sich leeren (`Minimum Pool Size = 0`), zählt für die Dimensionierung nur die Zahl der **gleichzeitig aktiven** Mandanten – nicht die Zahl aller angelegten.

3.3 Ändern

`Database:MaxPoolSizePerTenant` in der `appsettings.json` des API-Dienstes anpassen und den Dienst neu starten (die Grenze wird beim Verbindungsaufbau angewendet):

```
Restart-Service InfraDesk.API
```

4 Dimensionierung: die Formel

Die Client-Pools müssen zur Server-Obergrenze passen. Faustformel für die **theoretische Spitze**:

> **Empfehlung – Dimensionierungsregel:** `max_connections ≥ (gleichzeitig aktive Mandanten + 1) × MaxPoolSizePerTenant + Reserve`
> Das `+ 1` steht für die zentrale Registry-/Haupt-DB. Die **Reserve** (~15–20) deckt die für Superuser reservierten Verbindungen (`superuser_reserved_connections` , Standard 3), Sicherungen (`pg_dump`), das Monitoring und die manuelle Administration ab.

Nach dem Pool aufgelöst:

`MaxPoolSizePerTenant ≤ (max_connections − Reserve) ÷ (gleichzeitig aktive Mandanten + 1)`

Daraus ergeben sich **zwei Betriebsformen**:

- **Wenige große Mandanten** (Corporate/Kommune-Einzelmandant): großzügiger Pool, da wenige Pools den Server teilen.
- **Viele kleine Mandanten** (MSP, Kommune mit vielen Untermandanten): **kleiner** Pool, da sich viele Pools die Obergrenze teilen. Alternativ die Mandanten auf **mehrere DB-Server** verteilen.

Ein vorgeschalteter Pooler (z. B. PgBouncer) löst das bei Database-per-Tenant **nicht**: Er poolt je (Benutzer, Datenbank) – aus einem Pool würden **N** Pools. Dieselbe Wirkung erzielt die eine Zeile `MaxPoolSizePerTenant` . (PgBouncer ist nur in der optionalen HA-Topologie mit **einer** Haupt-DB sinnvoll.)

5 Empfohlene Einstellungen je Kundengröße (Lizenz)

Die PostgreSQL-Spalten sind die Werte, die der Installer auf **selbst installierten** Servern automatisch berechnet – zugleich die **Zielwerte**, wenn Sie eine externe Instanz von Hand einstellen (Abschnitt 6). Die Hardware-Angaben knüpfen an die Basislinie aus ID-02 an (Minimum 8 GB / 4 Kerne).

5.1 Corporate (Privatwirtschaft)

Ein Unternehmen = **ein** Mandant (Ausnahme: Konzern mit bewusst getrennten Untermandanten).

Lizenzstufe (Admins/Agenten)	RAM	Kerne	`max_connections`	`shared_buffers`	`effective_cache_size`	`work_mem`	MaxPoolSizePerTenant
SMB (5 / 2)	8 GB	4	100	2 GB	4 GB	10 MB	**30**

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Lizenzstufe (Admins/Agenten)	RAM	Kerne	`max_connections`	`shared_buffers`	`effective_cache_size`	`work_mem`	MaxPoolSizePerTenant
MID (20 / 5)	16 GB	4–8	128	4 GB	8 GB	16 MB	**50** (Standard)
Large (30 / 10)	32 GB	8	256	8 GB	16 GB	16 MB	**80**
Enterprise (∞ / ∞)	64 GB	16	400	8 GB[1]	32 GB	16 MB	**100** (1 Mandant)

[1] `shared_buffers` ist bei 8 GB gedeckelt. Auf einem **dedizierten** DB-Server (ohne API und Portal) darf er höher, und `effective_cache_size` auf ~75 % steigen.

> **Hinweis SMB:** Auf der 8-GB-Minimalausstattung ist `max_connections` nur 100. Der empfohlene Pool von **30** hält bewusst Reserve frei; bei der geringen Last eines SMB-Mandanten ist auch der Standard 50 unkritisch. **Enterprise mit mehreren Untermandanten:** den Pool nach der Formel aus Abschnitt 4 teilen (Richtwert **50** je Mandant bei bis zu ~6 gleichzeitig aktiven).

5.2 Kommunen (Gov)

Kommunen betreiben häufig **mehrere** Mandanten (Kernverwaltung, Eigenbetriebe, Verwaltungsgemeinschaft). Die Lizenz staffelt nach Einwohnern; für die Dimensionierung zählt die **Mandantenzahl**.

Größe (Einwohnerstufe)	RAM	Kerne	`max_connections`	gleichz. aktive Mandanten	MaxPoolSizePerTenant
Klein (Stufe 1–3)	8–16 GB	4	100–128	1–3	**25–30**
Mittel (Stufe 4–6)	16–32 GB	4–8	128–256	3–8	**20–25**
Groß (Stufe 7–9)	32–64 GB	8–16	256–400	5–15	**15–25** (nach Formel)

> **Empfehlung:** Den Pool aus `max_connections` und der **erwarteten Zahl gleichzeitig aktiver Mandanten** herleiten (Abschnitt 4), nicht raten. Beispiel Stufe 6: 32 GB → `max_connections` 256, ~8 aktive Mandanten → `(256 − 20) ÷ (8 + 1) ≈ 26` → **25**.

5.3 MSP (viele Kundenmandanten)

Der Managed-Service-Provider betreibt **viele** Kundenmandanten auf gemeinsamer Infrastruktur. Hier ist ein **kleiner** Pool entscheidend.

Szenario	RAM	Kerne	`max_connections`	gleichz. aktive Mandanten	MaxPoolSizePerTenant
Einstieg	32 GB	8	256	~20 (von z. B. 100 angelegten)	**8–10**
Wachstum	64 GB	16	400	~40	**7–8**

> **Achtung:** Bei sehr vielen Mandanten reicht ein einzelner Server nicht beliebig weit. Faustregel: Übersteigt die Zahl **gleichzeitig aktiver** Mandanten die durch `max_connections` tragbare Menge (bei kleinstem sinnvollen Pool ~5–6), die Mandanten auf **mehrere DB-Server/Knoten** verteilen. Nur gleichzeitig aktive zählen – leerlaufende Pools geben ihre Verbindungen frei (Abschnitt 3.2).

6 Mitgebrachte / externe PostgreSQL-Instanz

> 🔑 Datenbank-Administrator auf der PostgreSQL-Instanz — nicht in Client oder Portal, sondern unmittelbar an der Datenbank (`psql` oder ein gleichwertiges Werkzeug).

Ist im Setup eine **externe** Datenbank gewählt oder läuft bereits ein PostgreSQL, tunt der Installer diese Instanz **nicht** – ihre Konfiguration gehört Ihnen.

🔒 **Pflichtreihenfolge** — `max_connections` und `shared_buffers` greifen **erst nach einem Neustart** des Dienstes. Wer ihn auslässt, misst danach die alten Werte und sucht am falschen Ende.

1. Ermitteln Sie die Größenstufe Ihrer Instanz nach Abschnitt 5.
2. Setzen Sie die Zielwerte, hier am Beispiel der Stufe *Large* (32 GB):

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
ALTER SYSTEM SET max_connections = '256';
ALTER SYSTEM SET shared_buffers = '8GB';
ALTER SYSTEM SET effective_cache_size = '16GB';
ALTER SYSTEM SET work_mem = '16MB';
ALTER SYSTEM SET maintenance_work_mem = '2GB';
ALTER SYSTEM SET idle_in_transaction_session_timeout = '60s';
-- danach: Dienst neu starten (max_connections/shared_buffers brauchen das)
```

`MaxPoolSizePerTenant` wird unabhängig davon **immer** in der `appsettings.json` gesetzt – die Client-Grenze ist keine Server-Einstellung.

> **Wichtig:** Achten Sie darauf, dass `max_connections` der **externen** Instanz zur Summe Ihrer Pools passt (Formel in Abschnitt 4). Teilt sich die Instanz mit **anderen** Anwendungen den Server, deren Verbindungen von Ihrem Budget abziehen.

7 Prüfen & Überwachen

Aktuelle Werte am Server auslesen:

```
SELECT name, setting, unit FROM pg_settings
WHERE name IN ('max_connections','shared_buffers','effective_cache_size','work_mem','max_wal_size');

-- Momentane Verbindungslast (gesamt gegen die Obergrenze)
SELECT count(*) AS aktiv,
       current_setting('max_connections')::int AS obergrenze
FROM pg_stat_activity;

-- Verbindungen je Datenbank (= je Mandant)
SELECT datname, count(*) FROM pg_stat_activity GROUP BY datname ORDER BY 2 DESC;
```

Im Produkt: Die SuperAdmin-**Plattformüberwachung** zeigt die Verbindungslast in Prozent der `max_connections`. Standard-Warnschwellen (über `Platform:Thresholds` anpassbar):

Schwelle	Warnung	Kritisch
Verbindungslast (% von `max_connections`)	**80 %**	**90 %**
Plattenfüllstand	85 %	92 %
Langlaufende Abfragen (> 30 s)	ab 3	–

Erreicht die Verbindungslast dauerhaft die Warnschwelle, ist entweder der Pool zu groß für die Mandantenzahl oder `max_connections` (bzw. die Hardware) zu klein – siehe Formel in Abschnitt 4.

8 Symptome & Diagnose

Symptom	Ursache	Abhilfe
`FATAL: sorry, too many clients already` / `remaining connection slots are reserved`	`max_connections` ausgeschöpft: Zahl der Mandanten × Pool zu hoch.	`MaxPoolSizePerTenant` senken **oder** `max_connections` /RAM erhöhen (Formel Abschnitt 4).
Client-Zeitüberschreitungen beim Verbindungsaufbau, obwohl der Server nicht ausgelastet ist	Pool eines einzelnen Mandanten zu **klein** für seine Lastspitze.	`MaxPoolSizePerTenant` erhöhen (sofern `max_connections` Reserve hat).
Plattform-Monitoring meldet Verbindungslast > **80 %**	Nähert sich der Obergrenze.	Frühwarnung – Dimensionierung prüfen, bevor es zu Ablehnungen kommt.
Server wird beim Start/Update langsam, viel Speicherverbrauch	`max_connections` × `work_mem` zu hoch für den RAM.	`work_mem` prüfen (wird je Sortier-/Hash-Schritt je Verbindung belegt), ggf. senken.
Nach Änderung von `max_connections` keine Wirkung	Nur `reload` statt Neustart ausgeführt.	Dienst `postgresql-ordivis` neu starten (Abschnitt 2.2).

Verwandte Themen

- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Installation und Betrieb der Datenbank

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

- **ID-02 – Systemvoraussetzungen & Architektur** – Hardware- und Netzvoraussetzungen
- **ID-52 – Datenbank-Referenz** – Tabellen, Indizes und Feldstruktur
- **ID-27a – Reports & SQL-Abfragen** – Abfragen, die Last erzeugen

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TECHNISCHE REFERENZ

Datenbank-Referenz

Datenbankstruktur und Felddefinitionen für SQL-Reports

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-52 Datenbank-Referenz

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Datenbank-Referenz

Diese Referenz dokumentiert die Datenstruktur von Ordivis Platform für Auswertungen und **eigene SQL-Abfragen** im Report-Baukasten (ID-27a – *Reports & SQL-Abfragen*). Sie besteht aus zwei Teilen:

- **Teil A – Report-Felder je Modul** (kuratiert): die im Baukasten per Drag & Drop verfügbaren Felder mit deutschen Definitionen und der zugrunde liegenden Datenbankspalte. Ideal, um schnell die richtige Spalte für einen Report zu finden.
- **Teil B – Vollständiges Datenbankschema**: jede Tabelle und Spalte mit Datentyp – die vollständige Grundlage für Roh-SQL. Katalog-Felder aus Teil A sind hier zusätzlich mit ihrer Bedeutung angereichert.

Wichtig: Alle Tabellen der Fachmodule sind mandantenfähig und tragen eine Spalte `tenant_id`. Eigene SQL-Abfragen **müssen** ihre Ergebnisse mit `WHERE … tenant_id = @tenantId` auf den eigenen Mandanten einschränken – der Report-Baukasten setzt diesen Parameter automatisch. Zugriffe sind ausschließlich lesend.

Hinweis: Diese Referenz wird automatisch erzeugt (`Docs/manual/build/gen_db_reference.py`) – Teil A aus dem Feldkatalog des Baukastens, Teil B aus der Datenbank. Nicht von Hand pflegen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

INHALT

Teil A – Report-Felder je Modul
- CMDB
- Service-Desk
- IPAM
- Assets & Verträge
- Security / ISMS
- BCM
- Datenschutz
- Identität
- Checklisten
- Audit

Teil B – Vollständiges Datenbankschema
- Schema activation
- Schema alerting
- Schema audit
- Schema backup
- Schema bcm
- Schema cmdb
- Schema contracts
- Schema datenschutz
- Schema discovery
- Schema drplanning
- Schema identity
- Schema ipam
- Schema jobs
- Schema licensing
- Schema notifications
- Schema public
- Schema registry
- Schema reporting
- Schema security
- Schema selforg
- Schema ticketing
- Schema vulncatalog
- Schema workflow

Verwandte Themen

Teil A – Report-Felder je Modul

CMDB

Configuration Items (CMDB) (`cmdb.ci`) – Alle Configuration Items mit Klasse und kaufmännischen Daten.

Feld	Typ	Datenbankspalte	Bedeutung
CI-Nummer	Text	`cmdb.configuration_items.ci_number`	Eindeutige, schema-erzeugte CI-Kennung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Name	Text	`cmdb.configuration_items.name`	Anzeigename des CI.
CI-Klasse	Text	`cmdb.ci_classes.name`	Klasse/Typ des CI (z. B. Server, Anwendung).
Lebenszyklus	Auswahl	`cmdb.configuration_items.lifecycle_status`	Lebenszyklus-Status des CI.
Seriennummer	Text	`cmdb.configuration_items.serial_number`	Hersteller-Seriennummer.
Hersteller	Text	`cmdb.configuration_items.manufacturer`	Hersteller.
Modell	Text	`cmdb.configuration_items.model`	Modell-/Typbezeichnung.
Kostenstelle	Text	`cmdb.configuration_items.cost_center`	Kostenstelle.
Kaufpreis	Zahl	`cmdb.configuration_items.purchase_price`	Anschaffungspreis.
Kaufdatum	Datum	`cmdb.configuration_items.purchase_date`	Anschaffungsdatum.
Garantieende	Datum	`cmdb.configuration_items.warranty_end`	Ende der Herstellergarantie.
Support-Ende	Datum	`cmdb.configuration_items.support_end`	Ende des Supportvertrags.
Zuletzt bestätigt	Datum	`cmdb.configuration_items.last_verified_at`	Letzte inhaltliche Bestätigung des CI.
Geändert am	Datum	`cmdb.configuration_items.modified_at`	Letzte Änderung.

Service-Desk

Tickets (Service-Desk) (`ticketing.tickets`) – Störungen, Anfragen, Probleme und Änderungen mit Status, Priorität und SLA.

Feld	Typ	Datenbankspalte	Bedeutung
Ticket-Nr.	Text	`ticketing.tickets.ticket_number`	Eindeutige Ticketnummer.
Betreff	Text	`ticketing.tickets.subject`	Betreff/Kurzbeschreibung.
Typ	Auswahl	`ticketing.tickets.type`	Ticket-Art.
Status	Auswahl	`ticketing.tickets.status`	Bearbeitungsstatus.
Priorität	Auswahl	`ticketing.tickets.priority`	Priorität.
Kategorie	Text	`ticketing.tickets.category1`	Hauptkategorie.
Unterkategorie	Text	`ticketing.tickets.category2`	Unterkategorie.
Team	Text	`ticketing.tickets.support_group_name`	Zugewiesenes Support-Team.
Melder	Text	`ticketing.tickets.reporter_email`	E-Mail des Melders.
Eskalationsstufe	Zahl	`ticketing.tickets.escalation_level`	0 = keine Eskalation.
Bekannter Fehler	Ja/Nein	`ticketing.tickets.is_known_error`	Als bekannter Fehler markiert.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Genehmigung	Text	`ticketing.tickets.approval_status`	Genehmigungsstatus (z. B. Pending/Approved).
Erstellt am	Datum	`ticketing.tickets.created_at`	Erstellzeitpunkt.
SLA-Frist	Datum	`ticketing.tickets.sla_due_at`	Fällig laut Lösungs-SLA.
Erstreaktion	Datum	`ticketing.tickets.first_responded_at`	Zeitpunkt der Erstreaktion.
SLA-Verletzung	Datum	`ticketing.tickets.sla_response_breached_at`	Zeitpunkt einer Reaktions-SLA-Verletzung.

Service-Desk-Kennzahlen je Tag (`ticketing.snapshots`) – Tägliche Kennzahlen: Aufkommen, Lösungen, Backlog und SLA-Erfüllung.

Feld	Typ	Datenbankspalte	Bedeutung
Tag	Datum	`ticketing.ticket_metric_snapshots.day`	Kennzahlen-Tag.
Neu erstellt	Zahl	`ticketing.ticket_metric_snapshots.created_count`	Am Tag erstellte Tickets.
Gelöst	Zahl	`ticketing.ticket_metric_snapshots.resolved_count`	Am Tag gelöste Tickets.
Offener Rückstand	Zahl	`ticketing.ticket_metric_snapshots.open_backlog`	Offene Tickets am Tagesende.
Lösungs-SLA erfüllt	Zahl	`ticketing.ticket_metric_snapshots.resolution_sla_met`	Innerhalb Lösungs-SLA.
Lösungs-SLA gesamt	Zahl	`ticketing.ticket_metric_snapshots.resolution_sla_total`	Bewertete Lösungs-SLAs.
Reaktions-SLA erfüllt	Zahl	`ticketing.ticket_metric_snapshots.response_sla_met`	Innerhalb Reaktions-SLA.
Reaktions-SLA gesamt	Zahl	`ticketing.ticket_metric_snapshots.response_sla_total`	Bewertete Reaktions-SLAs.

IPAM

IP-Adressen (IPAM) (`ipam.ip`) – Vergebene und erkannte IP-Adressen inkl. Netz und Scan-Herkunft.

Feld	Typ	Datenbankspalte	Bedeutung
IP-Adresse	Text	`ipam.ip_addresses.address`	IP-Adresse.
Netz	Text	`ipam.prefixes.network`	Zugehöriges Netz (Prefix).
Status	Auswahl	`ipam.ip_addresses.status`	Vergabestatus.
DNS-Name	Text	`ipam.ip_addresses.dns_name`	Hinterlegter DNS-Name.
Beschreibung	Text	`ipam.ip_addresses.description`	Freitext.
Scan-Hostname	Text	`ipam.ip_addresses.scanned_hostname`	Beim Scan ermittelter Hostname.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
MAC-Adresse	Text	`ipam.ip_addresses.scanned_mac_address`	Beim Scan ermittelte MAC.
Unbestätigt	Ja/Nein	`ipam.ip_addresses.is_unconfirmed`	Scan-Fund, noch nicht bestätigt.
Zuletzt gesehen	Datum	`ipam.ip_addresses.last_seen_at`	Letzter Scan-Kontakt.

Netze / Prefixe (IPAM) (`ipam.prefix`) – Netze inkl. DHCP-Konfiguration und letztem Scan.

Feld	Typ	Datenbankspalte	Bedeutung
Netz	Text	`ipam.prefixes.network`	Netz in CIDR-Schreibweise.
Status	Auswahl	`ipam.prefixes.status`	Status des Netzes.
Beschreibung	Text	`ipam.prefixes.description`	Freitext.
DHCP aktiv	Ja/Nein	`ipam.prefixes.dhcp_enabled`	DHCP für dieses Netz aktiv.
DHCP-Bereich von	Text	`ipam.prefixes.dhcp_range_start`	Erste DHCP-Adresse.
DHCP-Bereich bis	Text	`ipam.prefixes.dhcp_range_end`	Letzte DHCP-Adresse.
Zuletzt gefunden	Zahl	`ipam.prefixes.last_scan_found_count`	Trefferzahl des letzten Scans.
Letzter Scan	Datum	`ipam.prefixes.last_scan_at`	Zeitpunkt des letzten Scans.

Assets & Verträge

Verträge (`contracts.contracts`) – Verträge mit Anbieter, Laufzeit, Kosten und Kündigungsfristen.

Feld	Typ	Datenbankspalte	Bedeutung
Titel	Text	`contracts.contracts.title`	Vertragsbezeichnung.
Anbieter	Text	`contracts.contracts.vendor_name`	Vertragspartner.
Vertragsnummer	Text	`contracts.contracts.contract_number`	Vertragsnummer.
Art	Text	`contracts.contracts.type`	Vertragsart.
Status	Text	`contracts.contracts.status`	Vertragsstatus.
Beginn	Datum	`contracts.contracts.start_date`	Vertragsbeginn.
Ende	Datum	`contracts.contracts.end_date`	Vertragsende.
Jahreskosten	Zahl	`contracts.contracts.annual_cost`	Jährliche Kosten.
Kündigungsfrist (Tage)	Zahl	`contracts.contracts.notice_period_days`	Kündigungsfrist in Tagen.
Kostenstelle	Text	`contracts.contracts.cost_center`	Kostenstelle.
Kündigung wirksam	Datum	`contracts.contracts.cancellation_effective_date`	Datum, zu dem die Kündigung wirkt.

Lizenzen (`licensing.entitlements`) – Lizenz-Ansprüche mit Menge, Kosten und Ablauf.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Produkt	Text	`licensing.license_entitlements.product_name`	Produktname.
Hersteller	Text	`licensing.license_entitlements.publisher`	Softwarehersteller.
Lizenztyp	Text	`licensing.license_entitlements.license_type`	Art der Lizenz.
Status	Text	`licensing.license_entitlements.status`	Lizenzstatus.
Gekaufte Menge	Zahl	`licensing.license_entitlements.purchased_quantity`	Erworbene Anzahl.
Stückkosten	Zahl	`licensing.license_entitlements.cost_per_unit`	Kosten je Einheit.
Ablauf	Datum	`licensing.license_entitlements.expires_at`	Ablaufdatum der Lizenz.
Support-Ende	Datum	`licensing.license_entitlements.support_ends_at`	Ende des Herstellersupports.
Vertragsbezug	Text	`licensing.license_entitlements.contract_reference`	Referenz auf den Vertrag.

Security / ISMS

SoA / Kontrollen (ISMS) (`security.soa`) – Erklärung zur Anwendbarkeit (Statement of Applicability) mit Umsetzungsstatus.

Feld	Typ	Datenbankspalte	Bedeutung
Kontroll-Nr.	Text	`security.soa_entries.control_external_id`	Kontroll-Kennung (z. B. A.5.1).
Kontrolle	Text	`security.soa_entries.control_title`	Titel der Kontrolle.
Anwendbar	Ja/Nein	`security.soa_entries.is_applicable`	Kontrolle ist anwendbar.
Umsetzungsstatus	Auswahl	`security.soa_entries.implementation_status`	Soll-Ist-Umsetzungsstatus.
Verantwortlich	Text	`security.soa_entries.responsible_party`	Verantwortliche Stelle.
Begründung	Text	`security.soa_entries.justification`	Begründung zu Anwendbarkeit/Ausschluss.

Risiken (ISMS) (`security.risks`) – Bewertete Informationssicherheits-Risiken inkl. Restrisiko und Strategie.

Feld	Typ	Datenbankspalte	Bedeutung
Titel	Text	`security.security_risks.title`	Risikobezeichnung.
Kategorie	Text	`security.security_risks.category`	Risikokategorie (z. B. Hoch).
Eintritt	Zahl	`security.security_risks.likelihood`	Eintrittswahrscheinlichkeit.
Auswirkung	Zahl	`security.security_risks.impact`	Schadenshöhe.
Risikowert	Zahl	`security.security_risks.score`	Bruttorisiko (Wahrscheinlichkeit × Auswirkung).
Strategie	Text	`security.security_risks.strategy`	Behandlungsstrategie.
Restrisiko	Zahl	`security.security_risks.residual_score`	Risiko nach Maßnahmen.
Status	Text	`security.security_risks.status`	Bearbeitungsstatus.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Bedrohung	Text	`security.security_risks.threat_title`	Zugrunde liegende Bedrohung.

Nichtkonformitäten (ISMS) (`security.nonconformities`) – Abweichungen und Korrekturmaßnahmen mit Fälligkeit und Status.

Feld	Typ	Datenbankspalte	Bedeutung
Beschreibung	Text	`security.isms_nonconformities.description`	Beschreibung der Abweichung.
Quelle	Text	`security.isms_nonconformities.source`	Herkunft (z. B. Audit).
Status	Auswahl	`security.isms_nonconformities.status`	Bearbeitungsstand.
Fällig	Datum	`security.isms_nonconformities.due_date`	Fälligkeit der Korrekturmaßnahme.
Korrekturmaßnahme	Text	`security.isms_nonconformities.corrective_action`	Geplante Korrektur.

BCM

BCM-Prozessprofile (`bcm.process`) – Kritische Geschäftsprozesse mit RTO/RPO/MTPD und aktueller Beeinträchtigung.

Feld	Typ	Datenbankspalte	Bedeutung
Prozess	Text	`cmdb.configuration_items.name`	Prozessname (aus dem verknüpften CI).
MTPD (h)	Zahl	`bcm.bcm_process_profiles.mtpd_hours`	Maximal tolerierbare Ausfallzeit in Stunden.
RTO (h)	Zahl	`bcm.bcm_process_profiles.rto_hours`	Wiederanlaufzeit-Ziel in Stunden.
RPO (h)	Zahl	`bcm.bcm_process_profiles.rpo_hours`	Tolerierbarer Datenverlust in Stunden.
Beeinträchtigung	Zahl	`bcm.bcm_process_profiles.impairment`	Aktuelle Beeinträchtigungsstufe.
Mindestbetrieb	Text	`bcm.bcm_process_profiles.mbco`	Minimal aufrechtzuerhaltender Betrieb (MBCO).

BCM-Übungen (`bcm.exercise`) – Notfall-/Wiederanlaufübungen mit Planung und Durchführung.

Feld	Typ	Datenbankspalte	Bedeutung
Titel	Text	`bcm.bcm_exercises.title`	Übungsbezeichnung.
Art	Text	`bcm.bcm_exercises.type`	Übungsart.
Status	Text	`bcm.bcm_exercises.status`	Status der Übung.
Geplant	Datum	`bcm.bcm_exercises.planned_date`	Geplantes Datum.
Durchgeführt	Datum	`bcm.bcm_exercises.conducted_date`	Tatsächliche Durchführung.
Erkenntnisse	Text	`bcm.bcm_exercises.lessons_learned`	Lessons Learned.

BCM-Krisenereignisse (`bcm.crisis`) – Historie ausgerufener Krisen mit Höchststufe und Abschluss.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Beginn	Datum	`bcm.bcm_crisis_events.started_at`	Beginn der Krise.
Ende	Datum	`bcm.bcm_crisis_events.ended_at`	Ende der Krise.
Höchststufe	Zahl	`bcm.bcm_crisis_events.peak_level`	Höchste erreichte Eskalationsstufe.
Abgeschlossen von	Text	`bcm.bcm_crisis_events.closed_by`	Wer die Krise geschlossen hat.
Notiz	Text	`bcm.bcm_crisis_events.note`	Anmerkung.

Datenschutz

Verarbeitungstätigkeiten (VVT) (`datenschutz.processing`) – Verzeichnis der Verarbeitungstätigkeiten nach Art. 30 DSGVO.

Feld	Typ	Datenbankspalte	Bedeutung
Bezeichnung	Text	`datenschutz.processing_activities.name`	Name der Verarbeitung.
Zweck	Text	`datenschutz.processing_activities.purpose`	Zweck der Verarbeitung.
Abteilung	Text	`datenschutz.processing_activities.department`	Verantwortliche Abteilung.
Status	Text	`datenschutz.processing_activities.status`	Status des VVT-Eintrags.
Nächste Prüfung	Datum	`datenschutz.processing_activities.next_review_due_at`	Fälligkeit der nächsten Überprüfung.
Prüfintervall (Monate)	Zahl	`datenschutz.processing_activities.review_interval_months`	Überprüfungsintervall in Monaten.
Freigegeben am	Datum	`datenschutz.processing_activities.approved_at`	Zeitpunkt der Freigabe.

Datenpannen (`datenschutz.breach`) – Datenschutzverletzungen mit Risiko, Meldung und Betroffenenzahl.

Feld	Typ	Datenbankspalte	Bedeutung
Aktenzeichen	Text	`datenschutz.data_breaches.reference`	Referenz/Aktenzeichen.
Titel	Text	`datenschutz.data_breaches.title`	Kurzbeschreibung.
Status	Text	`datenschutz.data_breaches.status`	Bearbeitungsstatus.
Risiko	Text	`datenschutz.data_breaches.risk`	Risikoeinstufung.
Bekannt seit	Datum	`datenschutz.data_breaches.became_known_at`	Zeitpunkt der Kenntnis.
An Behörde gemeldet	Datum	`datenschutz.data_breaches.reported_to_authority_at`	Meldung an die Aufsichtsbehörde.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Betroffene	Zahl	`datenschutz.data_breaches.affected_subject_count`	Anzahl betroffener Personen.

Löschfristen (`datenschutz.retention`) – Aufbewahrungs-/Löschregeln je Verarbeitungstätigkeit.

Feld	Typ	Datenbankspalte	Bedeutung
Verarbeitung	Text	`datenschutz.processing_activities.name`	Zugehörige Verarbeitung.
Aufbewahrung (Monate)	Zahl	`datenschutz.retention_rules.retention_months`	Aufbewahrungsdauer in Monaten.
Auslöser	Text	`datenschutz.retention_rules.trigger`	Fristbeginn-Auslöser.
Aktion	Text	`datenschutz.retention_rules.action`	Aktion nach Fristablauf (z. B. Löschen).
Rechtsgrundlage	Text	`datenschutz.retention_rules.legal_reference`	Gesetzliche Grundlage.

Identität

Benutzerkonten (`identity.users`) – Benutzerkonten mit Status, Abteilung, MFA und letzter Anmeldung.

Feld	Typ	Datenbankspalte	Bedeutung
Anmeldename	Text	`identity.users.login`	Anmeldename.
Anzeigename	Text	`identity.users.display_name`	Anzeigename.
E-Mail	Text	`identity.users.email`	E-Mail-Adresse.
Status	Auswahl	`identity.users.status`	Kontostatus.
Abteilung	Text	`identity.users.department`	Abteilung.
Funktion	Text	`identity.users.job_title`	Position/Funktion.
MFA-Pflicht	Ja/Nein	`identity.users.mfa_enforced`	Mehr-Faktor-Authentisierung erzwungen.
Systemkonto	Ja/Nein	`identity.users.is_system_account`	Technisches/Systemkonto.
Letzte Anmeldung	Datum	`identity.users.last_login_at`	Zeitpunkt der letzten Anmeldung.
Gesperrt bis	Datum	`identity.users.locked_until`	Sperre bis (Fehlanmeldungen).
Angelegt am	Datum	`identity.users.created_at`	Anlagezeitpunkt.

Rollenzuweisungen (`identity.user_roles`) – Welcher Benutzer trägt welche Rolle (mit Gültigkeit).

Feld	Typ	Datenbankspalte	Bedeutung
Rolle	Text	`identity.roles.name`	Rollenname.
Anmeldename	Text	`identity.users.login`	Anmeldename des Trägers.
Benutzer	Text	`identity.users.display_name`	Anzeigename des Trägers.
Abteilung	Text	`identity.users.department`	Abteilung des Trägers.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Feld	Typ	Datenbankspalte	Bedeutung
Gültig ab	Datum	`identity.user_roles.valid_from`	Beginn der Gültigkeit.
Gültig bis	Datum	`identity.user_roles.valid_until`	Ende der Gültigkeit.

Checklisten

Checklisten-Durchläufe (`checklist.runs`) – Aktive und abgeschlossene Checklisten-Durchführungen mit Fälligkeit.

Feld	Typ	Datenbankspalte	Bedeutung
Titel	Text	`bcm.checklist_template_runs.title`	Titel des Durchlaufs.
Status	Auswahl	`bcm.checklist_template_runs.status`	Status der Durchführung.
Zugewiesen an	Text	`bcm.checklist_template_runs.assigned_to_name`	Verantwortliche Person.
Fällig	Datum	`bcm.checklist_template_runs.due_at`	Fälligkeitszeitpunkt.
Gestartet	Datum	`bcm.checklist_template_runs.started_at`	Startzeitpunkt.
Abgeschlossen	Datum	`bcm.checklist_template_runs.completed_at`	Abschlusszeitpunkt.
Auslöser	Text	`bcm.checklist_template_runs.trigger_type`	Wodurch ausgelöst.

Checklisten-Vorlagen (`checklist.templates`) – Vorlagenkatalog mit Bereich, Version und Auto-Auslöser.

Feld	Typ	Datenbankspalte	Bedeutung
Titel	Text	`bcm.checklist_templates.title`	Vorlagentitel.
Bereich	Text	`bcm.checklist_templates.category`	Fachbereich/Kategorie.
Version	Zahl	`bcm.checklist_templates.version`	Vorlagenversion.
Aktiv	Ja/Nein	`bcm.checklist_templates.is_active`	Vorlage aktiv.
Auto-Auslöser	Text	`bcm.checklist_templates.auto_trigger`	Regelbasierter Auslöser.

Audit

Audit-Protokoll (`audit.entries`) – Revisionssichere Ereignisse: wer hat wann was geändert.

Feld	Typ	Datenbankspalte	Bedeutung
Zeitpunkt	Datum	`audit.audit_entries.occurred_on`	Ereigniszeitpunkt.
Benutzer	Text	`audit.audit_entries.user_name`	Auslösender Benutzer.
Aktion	Text	`audit.audit_entries.action`	Durchgeführte Aktion.
Objekttyp	Text	`audit.audit_entries.entity_type`	Betroffener Objekttyp.
IP-Adresse	Text	`audit.audit_entries.ip_address`	Quell-IP.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Teil B – Vollständiges Datenbankschema

Schema `activation`

activation.activation_nodes

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`license_id`	Text	•	
`fingerprint`	Text	•	
`first_seen_utc`	Zeitstempel	•	
`last_seen_utc`	Zeitstempel	•	

activation.activation_states

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`trial_start_utc`	Zeitstempel	•	
`last_seen_utc`	Zeitstempel	•	
`fingerprint`	Text	•	
`integrity_hmac`	Text	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

activation.product_licenses

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`license_id`	Text	•	
`customer_name`	Text	•	
`sku`	Text	•	
`max_admins`	Ganzzahl	•	
`max_tenants`	Ganzzahl	•	
`issued_at`	Zeitstempel	•	
`expiration_date`	Zeitstempel	•	
`bound_fingerprint`	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
raw_token	Text	•	
key_id	Text	•	
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
max_nodes	Ganzzahl	•	
max_agents	Ganzzahl	•	

Schema alerting

alerting.alert_acknowledgements

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
alert_id	UUID	•	
recipient_id	UUID	•	
response	Ganzzahl	•	
acknowledged_at	Zeitstempel	•	
note	Text		

alerting.alert_deliveries

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
alert_id	UUID	•	
recipient_id	UUID	•	
channel	Ganzzahl	•	
attempt	Ganzzahl	•	
state	Ganzzahl	•	
provider_code	Text		
detail	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
attempted_at	Zeitstempel	•	

alerting.alert_recipients

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
alert_id	UUID	•	
user_id	UUID		
contact_id	UUID		
display_name	Text	•	
escalation_level	Ganzzahl	•	
email	Text		

alerting.alerts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
title	Text	•	
body	Text	•	
severity	Ganzzahl	•	
status	Ganzzahl	•	
source	Ganzzahl	•	
triggered_at	Zeitstempel	•	
triggered_by_user_id	UUID		
related_entity_type	Text		
related_entity_id	UUID		
closed_at	Zeitstempel		
closed_by_user_id	UUID		

Schema audit

audit.audit_entries

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`tenant_id`	UUID	•	
`user_id`	UUID	•	
`user_name`	Text	•	Auslösender Benutzer.
`ip_address`	Text		Quell-IP.
`action`	Text	•	Durchgeführte Aktion.
`entity_type`	Text	•	Betroffener Objekttyp.
`entity_id`	UUID		
`delta`	Text		
`correlation_id`	UUID		
`occurred_on`	Zeitstempel	•	Ereigniszeitpunkt.
`source`	Text		
`device_id`	Text		

Schema `backup`

backup.backup_findings

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`verification_id`	UUID	•	
`run_id`	UUID	•	
`cause`	Text	•	
`action`	Text		
`status`	Ganzzahl	•	
`follow_up_at`	Zeitstempel		
`ticket_id`	UUID		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

backup.backup_job_sources

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
job_id	UUID	•	
ci_id	UUID	•	
scope	Text		
relation_mirrored	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_job_tiers

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
job_id	UUID	•	
sequence	Ganzzahl	•	
name	Text	•	
target_id	UUID	•	
source_tier_id	UUID		
uses_own_schedule	Ja/Nein	•	
recurrence_kind	Ganzzahl	•	
recurrence_weekdays_mask	Ganzzahl	•	
recurrence_day_of_month	Ganzzahl	•	
recurrence_month	Ganzzahl	•	
recurrence_nth	Ganzzahl	•	
recurrence_weekday	Ganzzahl	•	
recurrence_start_time	time without time zone	•	
recurrence_time_zone_id	Text	•	
recurrence_window_minutes	Ganzzahl	•	
recurrence_tolerance_minutes	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
retention_daily	Ganzzahl	●	
retention_weekly	Ganzzahl	●	
retention_monthly	Ganzzahl	●	
retention_yearly	Ganzzahl	●	
monitored	Ja/Nein	●	
created_at	Zeitstempel	●	
created_by	UUID	●	
modified_at	Zeitstempel	●	
modified_by	UUID	●	

backup.backup_jobs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	●	
tenant_id	UUID	●	
policy_id	UUID	●	
name	Text	●	
external_name	Text		
backup_system_ci_id	UUID		
method	Ganzzahl	●	
recurrence_kind	Ganzzahl	●	
recurrence_weekdays_mask	Ganzzahl	●	
recurrence_day_of_month	Ganzzahl	●	
recurrence_month	Ganzzahl	●	
recurrence_nth	Ganzzahl	●	
recurrence_weekday	Ganzzahl	●	
recurrence_start_time	time without time zone	●	
recurrence_time_zone_id	Text	●	
recurrence_window_minutes	Ganzzahl	●	
recurrence_tolerance_minutes	Ganzzahl	●	
owner_user_id	UUID		
active_from	Zeitstempel		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
active_to	Zeitstempel		
is_active	Ja/Nein	•	
notes	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_mail_inbox

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
sender	Text		
subject	Text	•	
received_at	Zeitstempel	•	
excerpt	Text		
handled	Ja/Nein	•	
resulting_run_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_mail_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
job_id	UUID	•	
tier_id	UUID		
name	Text	•	
sender_pattern	Text		
subject_pattern	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
success_pattern	Text		
failure_pattern	Text		
warning_pattern	Text		
use_regex	Ja/Nein	•	
is_active	Ja/Nein	•	
match_count	Ganzzahl	•	
last_match_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_media

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
pool_id	UUID	•	
barcode	Text	•	
label	Text		
state	Ganzzahl	•	
write_cycles	Ganzzahl	•	
max_write_cycles	Ganzzahl		
retention_until	Zeitstempel		
last_written_at	Zeitstempel		
current_location_ci_id	UUID		
off_site	Ja/Nein	•	
last_read_check_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

backup.backup_media_movements

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
medium_id	UUID	•	
direction	Ganzzahl	•	
occurred_at	Zeitstempel	•	
location_ci_id	UUID		
handed_over_by	UUID		
received_by	UUID		
received_by_name	Text		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_media_pools

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
rotation	Ganzzahl	•	
description	Text		
drive_ci_id	UUID		
target_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_policies

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
protection_class	Ganzzahl	•	
owner_user_id	UUID		
deputy_user_id	UUID		
verification_rhythm	Ganzzahl	•	
verification_due_days	Ganzzahl	•	
target_rpo_hours	Ganzzahl		
target_rto_hours	Ganzzahl		
dr_plan_id	UUID		
legal_retention_years	Ganzzahl		
vvt_processing_id	UUID		
restore_test_interval_days	Ganzzahl	•	
soa_entry_id	UUID		
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_restore_tests

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
policy_id	UUID	•	
job_id	UUID		
tier_id	UUID		
kind	Ganzzahl	•	
performed_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
performed_by	UUID		
scope	Text	•	
outcome	Ganzzahl	•	
measured_rto_minutes	Ganzzahl		
measured_rpo_minutes	Ganzzahl		
notes	Text		
dr_test_result_id	UUID		
medium_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_run_daily

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
job_id	UUID	•	
tier_id	UUID		
day	Datum	•	
expected	Ganzzahl	•	
success	Ganzzahl	•	
warning	Ganzzahl	•	
failed	Ganzzahl	•	
missed	Ganzzahl	•	
suspended	Ganzzahl	•	
bytes	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

backup.backup_runs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
job_id	UUID	•	
tier_id	UUID		
scheduled_at	Zeitstempel	•	
started_at	Zeitstempel		
finished_at	Zeitstempel		
status	Ganzzahl	•	
origin	Ganzzahl	•	
bytes	Ganzzahl		
object_count	Ganzzahl		
message	Text		
external_run_id	Text		
medium_id	UUID		
evaluated_at	Zeitstempel		
ticket_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_settings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
compact_after_days	Ganzzahl	•	
purge_after_days	Ganzzahl	•	
ticket_subject_template	Text	•	
fallback_support_group_id	UUID		
auto_create_ticket_on_failure	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
ticket_reopen_window_days	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

backup.backup_targets

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
kind	Ganzzahl	•	
description	Text		
path	Text		
region	Text		
provider	Text		
data_in_eu	Ja/Nein		
location_ci_id	UUID		
target_ci_id	UUID		
capacity_tb	Dezimalzahl		
used_tb	Dezimalzahl		
immutability	Ganzzahl	•	
immutability_days	Ganzzahl		
air_gapped	Ja/Nein	•	
encrypted	Ja/Nein	•	
encryption_method	Text		
key_custody_reference	Text		
contract_id	UUID		
egress_cost_per_tb	Dezimalzahl		
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

backup.backup_verifications

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`policy_id`	UUID	•	
`period_start`	Zeitstempel	•	
`period_end`	Zeitstempel	•	
`due_at`	Zeitstempel	•	
`assignee_user_id`	UUID		
`deputy_user_id`	UUID		
`status`	Ganzzahl	•	
`completed_at`	Zeitstempel		
`completed_by`	UUID		
`completion_note`	Text		
`evidence_snapshot`	JSON		
`revision`	Ganzzahl	•	
`escalation_level`	Ganzzahl	•	
`last_reminder_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

Schema `bcm`

bcm.bcm_alert_contacts

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
scope_id	UUID	•	
contact_id	UUID		
manual_name	Text		
manual_email	Text		
manual_phone	Text		
role	Text	•	
emergency_phone	Text		
escalation_level	Ganzzahl	•	
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_bao_assignments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
role_id	UUID	•	
user_id	UUID		
contact_id	UUID		
manual_name	Text		
is_primary	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
presence	Ganzzahl	•	
presence_updated_at	Zeitstempel		

bcm.bcm_bao_roles

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
name	Text	•	
level	Text	•	
function	Text	•	
is_reporting_obligation	Ja/Nein	•	
requires_deputy	Ja/Nein	•	
stabs_sachgebiet	Text		
sort_order	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_bia_assessments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
ci_id	UUID	•	
is_time_critical	Ja/Nein	•	
reason	Text		
is_manual_override	Ja/Nein	•	
has_legal_obligation	Ja/Nein	•	
achievable_rto_hours	Ganzzahl		
review_interval_days	Ganzzahl	•	
last_reviewed_at	Zeitstempel		
notes	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_capa_actions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
exercise_id	UUID		
title	Text	•	
description	Text		
is_preventive	Ja/Nein	•	
responsible_name	Text		
due_date	Datum		
status	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_checklist_runs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
checklist_id	UUID	•	
title	Text	•	
started_at	Zeitstempel	•	
completed_at	Zeitstempel		
run_steps_json	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_checklists

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
title	Text	•	
steps_json	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_comm_channel_status

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
channel	Text	•	
status	Text	•	
detail	Text		
sort_order	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_crisis_events

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
scope_id	UUID	•	
started_at	Zeitstempel	•	Beginn der Krise.
ended_at	Zeitstempel	•	Ende der Krise.
peak_level	Ganzzahl	•	Höchste erreichte Eskalationsstufe.
note	Text		Anmerkung.
closed_by	Text	•	Wer die Krise geschlossen hat.
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_documents

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
title	Text	•	
doc_type	Text	•	
document_version	Text		
valid_from	Datum		
valid_until	Datum		
is_public	Ja/Nein	•	
source	Text	•	
notes	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
status	Text	•	

bcm.bcm_exercises

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
title	Text	•	Übungsbezeichnung.
type	Text	•	Übungsart.
status	Text	•	Status der Übung.
planned_date	Datum	•	Geplantes Datum.
conducted_date	Datum		Tatsächliche Durchführung.
lessons_learned	Text		Lessons Learned.
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_headcounts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
on_site	Ganzzahl	•	
evacuated	Ganzzahl	•	
missing	Ganzzahl	•	
injured	Ganzzahl	•	
responders	Ganzzahl	•	
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_log_entries

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
entry_at	Zeitstempel	•	
author_name	Text	•	
sachgebiet	Text		
category	Text	•	
text	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_process_profiles

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
ci_id	UUID	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
mtpd_hours	Ganzzahl		Maximal tolerierbare Ausfallzeit in Stunden.
rto_hours	Ganzzahl		Wiederanlaufzeit-Ziel in Stunden.
rpo_hours	Ganzzahl		Tolerierbarer Datenverlust in Stunden.
mbco	Text		Minimal aufrechtzuerhaltender Betrieb (MBCO).
immediate_measures	Text		
alerting_note	Text		
damage_matrix_json	Text		
impairment	Ganzzahl	•	Aktuelle Beeinträchtigungsstufe.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
impairment_updated_at	Zeitstempel		

bcm.bcm_resource_gauges

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
name	Text	•	
unit	Text		
current_value	double precision	•	
warn_threshold	double precision		
critical_threshold	double precision		
sort_order	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.bcm_scopes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
maturity_level	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
crisis_active	Ja/Nein	•	
crisis_activated_at	Zeitstempel		
crisis_note	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
crisis_level	Ganzzahl	•	
crisis_peak_level	Ganzzahl	•	

bcm.bcm_situation_items

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
title	Text	•	
description	Text		
category	Text	•	
priority	Text	•	
location	Text		
is_resolved	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
latitude	double precision		
longitude	double precision		

bcm.checklist_items

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
section_id	UUID	•	
title	Text	•	
description	Text		
is_mandatory	Ja/Nein	•	
responsible_role	Text		
expected_result	Text		
precondition	Text		
sort_order	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

bcm.checklist_run_items

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
run_id	UUID	•	
source_item_id	UUID	•	
section_title	Text	•	
title	Text	•	
description	Text		
is_mandatory	Ja/Nein	•	
responsible_role	Text		
sort_order	Ganzzahl	•	
status	Ganzzahl	•	
note	Text		
evidence_ref	Text		
done_by_user_id	UUID		
done_by_name	Text		
done_at	Zeitstempel		

bcm.checklist_sections

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
template_id	UUID	•	
title	Text	•	
sort_order	Ganzzahl	•	

bcm.checklist_template_runs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
template_id	UUID	•	
template_version	Ganzzahl	•	
title	Text	•	Titel des Durchlaufs.
status	Ganzzahl	•	Status der Durchführung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
assigned_to_user_id	UUID		
assigned_to_name	Text		Verantwortliche Person.
due_at	Zeitstempel		Fälligkeitszeitpunkt.
started_at	Zeitstempel	•	Startzeitpunkt.
completed_at	Zeitstempel		Abschlusszeitpunkt.
completed_by_signature	Text		
trigger_type	Text		Wodurch ausgelöst.
trigger_ref	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

bcm.checklist_templates

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
title	Text	•	Vorlagentitel.
description	Text		
category	Text		Fachbereich/Kategorie.
version	Ganzzahl	•	Vorlagenversion.
is_active	Ja/Nein	•	Vorlage aktiv.
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
auto_trigger	Text	•	Regelbasierter Auslöser.
trigger_config	Text		

bcm.handbook_blocks

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
chapter_id	UUID	•	
kind	Ganzzahl	•	
static_content	Text		
source_key	Text		
sort_order	Ganzzahl	•	

bcm.handbook_chapters

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
handbook_id	UUID	•	
parent_chapter_id	UUID		
title	Text	•	
sort_order	Ganzzahl	•	

bcm.handbooks

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
type	Ganzzahl	•	
title	Text	•	
description	Text		
version	Ganzzahl	•	
status	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
released_at	Zeitstempel		
released_content_hash	Text		
review_due_at	Zeitstempel		

Schema cmdb

cmdb.application_responsibilities

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
application_ci_id	UUID	•	
raci_type	Ganzzahl	•	
is_deputy	Ja/Nein	•	
user_id	UUID		
org_unit_ci_id	UUID		
position_id	UUID		
role_name	Text		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.asset_assignments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
assigned_to_user_id	UUID		
assigned_to_name	Text	•	
location	Text		
assigned_at	Zeitstempel	•	
handover_note	Text		
returned_at	Zeitstempel		
return_note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

cmdb.asset_disposals

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
method	Text	•	
disposed_on	Datum	•	
responsible_user_id	UUID		
data_erasure_confirmed	Ja/Nein	•	
data_erasure_method	Text		
proceeds_amount	Dezimalzahl		
residual_book_value	Dezimalzahl		
recipient	Text		
reference	Text		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.asset_lifecycle_events

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
phase	Text	•	
occurred_on	Datum	•	
responsible_user_id	UUID		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	

cmdb.asset_tag_schemes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
enabled	Ja/Nein	•	
segments_json	JSON	•	
separator	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.catalog_adoptions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
catalog_process_id	UUID	•	
state	Ganzzahl	•	
ci_id	UUID		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.catalog_editions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
key	Text	•	
name	Text	•	
publisher	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
edition	Text	•	
edition_sort_key	Text	•	
license_name	Text		
license_url	Text		
dataset_url	Text		
source_attribution	Text		
modifications	Text		
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.catalog_processes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
task_id	UUID	•	
product_id	UUID	•	
name	Text	•	
process_type	Ganzzahl	•	
level	Ganzzahl	•	
legal_basis	Text		
org_unit_suggestion	Text		
leika_ids	Text		
sort_order	Ganzzahl	•	
tenant_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
legal_basis_families	Text		

cmdb.catalog_products

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
parent_id	UUID		
level	Ganzzahl	•	
number	Text	•	
name	Text	•	
short_description	Text		
legal_mandate	Text		
target_groups	Text		
legal_character	Text		
sort_order	Ganzzahl	•	
tenant_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.catalog_raci

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope	Ganzzahl	•	
node_id	UUID	•	
raci_type	Ganzzahl	•	
user_id	UUID		
org_unit_ci_id	UUID		
position_id	UUID		
role_name	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
note	Text		
valid_from	Datum	•	
valid_to	Datum		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.catalog_tasks

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
product_id	UUID	•	
number	Text		
name	Text	•	
sort_order	Ganzzahl	•	
tenant_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ci_baselines

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
label	Text	•	
reason	Text		
snapshot	JSON	•	
created_at	Zeitstempel	•	
created_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ci_classes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
parent_class_id	UUID		
name	Text	•	Klasse/Typ des CI (z. B. Server, Anwendung).
description	Text		
icon	Text		
is_abstract	Ja/Nein	•	
attributes_schema	JSON	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ci_components

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
name	Text	•	
mac_address	Text		
component_type	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
peer_ci_id	UUID		
peer_component_id	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
speed_value	Ganzzahl		
speed_unit	Text		
kind	Ganzzahl	•	
label	Text		
description	Text		
sort_order	Ganzzahl	•	
enabled	Ja/Nein	•	
link_status	Ganzzahl	•	
mark_connected	Ja/Nein	•	
rear_port_component_id	UUID		
rear_port_position	Ganzzahl	•	
positions	Ganzzahl	•	
power_port_component_id	UUID		
feed_leg	Ganzzahl		
maximum_draw_watts	Ganzzahl		
allocated_draw_watts	Ganzzahl		
mgmt_only	Ja/Nein	•	
poe_mode	Ganzzahl		
poe_type	Text		
duplex	Ganzzahl		
position	Text		

cmdb.ci_nfc_tags

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
uid	Text	•	
note	Text		
has_signed_payload	Ja/Nein	•	
created_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ci_relations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_ci_id	UUID	•	
target_ci_id	UUID	•	
relation_type	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ci_tag_counters

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
context_key	Text	•	
value	Ganzzahl	•	

cmdb.ci_uhf_tags

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
epc	Text	•	
note	Text		
last_seen_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

cmdb.configuration_items

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`ci_class_id`	UUID	•	
`name`	Text	•	Anzeigename des CI.
`description`	Text		
`lifecycle_status`	Ganzzahl	•	Lebenszyklus-Status des CI.
`serial_number`	Text		Hersteller-Seriennummer.
`manufacturer`	Text		Hersteller.
`model`	Text		Modell-/Typbezeichnung.
`warranty_start`	Datum		
`warranty_end`	Datum		Ende der Herstellergarantie.
`support_contract`	Text		
`support_end`	Datum		Ende des Supportvertrags.
`purchase_date`	Datum		Anschaffungsdatum.
`purchase_price`	Dezimalzahl		Anschaffungspreis.
`cost_center`	Text		Kostenstelle.
`attributes`	JSON	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	Letzte Änderung.
`modified_by`	UUID	•	
`last_verified_at`	Zeitstempel		Letzte inhaltliche Bestätigung des CI.
`last_verified_by`	UUID		
`ci_number`	Text	•	Eindeutige, schema-erzeugte CI-Kennung.
`useful_life_months`	Ganzzahl		
`salvage_value`	Dezimalzahl		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
device_model_id	UUID		

cmdb.device_model_component_templates

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
device_model_id	UUID	•	
kind	Ganzzahl	•	
name	Text	•	
label	Text		
type	Text		
description	Text		
sort_order	Ganzzahl	•	
mgmt_only	Ja/Nein	•	
poe_mode	Ganzzahl		
poe_type	Text		
speed	Ganzzahl		
maximum_draw_watts	Ganzzahl		
allocated_draw_watts	Ganzzahl		
power_port_name	Text		
feed_leg	Ganzzahl		
rear_port_name	Text		
rear_port_position	Ganzzahl	•	
positions	Ganzzahl	•	
position	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.device_models

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
manufacturer	Text	•	
model	Text	•	
slug	Text	•	
part_number	Text		
height_half_units	Ganzzahl	•	
is_full_depth	Ja/Nein	•	
airflow	Ganzzahl		
subdevice_role	Ganzzahl	•	
weight	Dezimalzahl		
weight_unit	Ganzzahl		
weight_grams	Ganzzahl		
power_draw_watts	Ganzzahl		
exclude_from_utilization	Ja/Nein	•	
comments	Text		
front_image_path	Text		
rear_image_path	Text		
origin	Ganzzahl	•	
catalog_version	Text		
source_key	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.fim_master_processes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
key	Text	•	
object_key	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
version	Text	•	
detail_level	Text	•	
name	Text	•	
label	Text		
releasing_body	Text		
definition	Text		
description	Text		
trigger	Text		
result	Text		
keywords	Text		
process_class_id	Text		
process_class_name	Text		
participants	Text		
legal_bases	Text		
states	Text		
bpmn_xml	Text		
last_changed_on	Text		
approved_on	Text		
fetched_at	Zeitstempel	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.fim_process_classes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
key	Text	•	
object_key	Text	•	
name	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
area_id	Text		
area_name	Text		
product_id	Text		
product_name	Text		
bundle_id	Text		
bundle_name	Text		
kgst_comparison	Text		
kgst_id	Text		
kgst_note	Text		
changed_on	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.fim_services

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
key	Text	•	
object_key	Text	•	
level	Ganzzahl	•	
name	Text	•	
public_name	Text		
object_name	Text		
action_name	Text		
detail_name	Text		
type_raw	Text		
regulation_level	Ganzzahl	•	
execution_level	Ganzzahl	•	
synonyms	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
special_features	Text		
legal_basis	Text		
teaser	Text		
short_text	Text		
full_text	Text		
prerequisites	Text		
required_documents	Text		
costs	Text		
deadlines	Text		
procedure	Text		
processing_time	Text		
forms	Text		
notes	Text		
further_information	Text		
addressee	Text		
responsible_office	Text		
contact_point	Text		
portal_situations	Text		
portal_situation_codes	Text		
sdg_area	Text		
target_maturity	Text		
published_on	Text		
catalog_status	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.gvp_editions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
number	Ganzzahl	•	
title	Text	•	
effective_from	Datum	•	
issued_at	Zeitstempel	•	
issued_by	UUID	•	
issued_by_name	Text	•	
snapshot_json	JSON	•	
pdf_sha256	Text		
pdf	Binär		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.gvp_entries

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
org_unit_ci_id	UUID	•	
kind	Ganzzahl	•	
text	Text	•	
process_ci_id	UUID		
catalog_product_id	UUID		
position_id	UUID		
sort_order	Ganzzahl	•	
reason	Text		
valid_from	Datum	•	
valid_to	Datum		
created_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

cmdb.inventory_runs

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`scope_ci_class_id`	UUID		
`scope_location_ci_id`	UUID		
`status`	Ganzzahl	•	
`started_at`	Zeitstempel	•	
`started_by`	UUID	•	
`closed_at`	Zeitstempel		
`closed_by`	UUID		
`closing_note`	Text		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

cmdb.inventory_scans

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`run_id`	UUID	•	
`code`	Text	•	
`ci_id`	UUID		
`scanned_at`	Zeitstempel	•	
`scanned_by`	UUID	•	

cmdb.label_media

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
language	Ganzzahl	•	
width_mm	double precision	•	
height_mm	double precision	•	
dpi	Ganzzahl	•	
gap_mm	double precision	•	
darkness	Ganzzahl	•	
is_default	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.label_templates

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
barcode	Ganzzahl	•	
layout	Ganzzahl	•	
show_name	Ja/Nein	•	
show_serial	Ja/Nein	•	
show_class	Ja/Nein	•	
show_logo	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.netbox_class_mappings

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
kind	Ganzzahl	•	
source_slug	Text	•	
ci_class_id	UUID		
is_suggested	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.netbox_migrations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
base_url	Text	•	
api_token_protected	Text		
status	Ganzzahl	•	
detected_version	Text		
last_run_at	Zeitstempel		
last_run_summary	Text		
completed_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.netbox_object_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
object_type	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
source_id	Ganzzahl	•	
local_id	UUID	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.org_assignments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
position_id	UUID	•	
kind	Ganzzahl	•	
person_ci_id	UUID		
user_id	UUID		
display_name	Text	•	
share_percent	Ganzzahl	•	
valid_from	Datum	•	
valid_to	Datum		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.org_authorities

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
position_id	UUID	•	
operation_type	Text	•	
amount_limit	Dezimalzahl		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
requires_second_signature	Ja/Nein	•	
note	Text		
valid_from	Datum	•	
valid_to	Datum		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.org_functions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
org_unit_ci_id	UUID		
position_id	UUID		
role_ci_id	UUID		
name	Text	•	
person_ci_id	UUID		
holder_name	Text		
note	Text		
valid_from	Datum	•	
valid_to	Datum		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
deputy_person_ci_id	UUID		
deputy_name	Text		

cmdb.org_manual_editions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
number	Ganzzahl	•	
title	Text	•	
effective_from	Datum	•	
issued_at	Zeitstempel	•	
issued_by	UUID	•	
issued_by_name	Text		
snapshot_json	JSON	•	
pdf_sha256	Text		
pdf	Binär		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.org_positions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
org_unit_ci_id	UUID	•	
name	Text	•	
position_key	Text		
is_leadership	Ja/Nein	•	
fte	Dezimalzahl	•	
note	Text		
valid_from	Datum	•	
valid_to	Datum		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	
cost_center	Text		
functional_org_unit_ci_id	UUID		
purpose	Text		
tasks	Text		
authority	Text		
requirements	Text		

cmdb.ozg_leika_keys

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
service_id	UUID	•	
leika_id	Text	•	
name	Text	•	
type_raw	Text		
regulation_level	Ganzzahl	•	
execution_level	Ganzzahl	•	
sort_order	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ozg_service_mappings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
service_id	UUID	•	
scope	Ganzzahl	•	
node_id	UUID	•	
kind	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
origin	Ganzzahl	•	
via_leika_id	Text		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ozg_service_status

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
service_id	UUID	•	
maturity	Ganzzahl	•	
implementation	Ganzzahl	•	
portal_url	Text		
application_ci_id	UUID		
efa_provider	Text		
note	Text		
assessed_on	Datum	•	
assessed_by_name	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ozg_services

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
situation_id	UUID	•	
number	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
name	Text	•	
description	Text		
priority	Ganzzahl		
execution_scope	Ganzzahl	•	
sort_order	Ganzzahl	•	
tenant_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ozg_situation_references

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
situation_id	UUID	•	
service_id	UUID		
service_name	Text	•	
leika_id	Text		
primary_situation_name	Text		
sort_order	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.ozg_situations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
edition_id	UUID	•	
topic_id	UUID	•	
number	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`name`	Text	•	
`kind`	Ganzzahl	•	
`description`	Text		
`sort_order`	Ganzzahl	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

cmdb.ozg_topics

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`edition_id`	UUID	•	
`number`	Text	•	
`name`	Text	•	
`audience`	Ganzzahl	•	
`sort_order`	Ganzzahl	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

cmdb.process_data_objects

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`process_ci_id`	UUID	•	
`data_object_ci_id`	UUID	•	
`can_create`	Ja/Nein	•	
`can_read`	Ja/Nein	•	
`can_update`	Ja/Nein	•	
`can_delete`	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
is_master_system	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.process_definitions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
process_ci_id	UUID	•	
version	Ganzzahl	•	
name	Text	•	
bpmn_xml	Text	•	
status	Ganzzahl	•	
approved_by	UUID		
approved_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.process_kpis

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
process_ci_id	UUID	•	
name	Text	•	
description	Text		
unit	Text		
target_value	Dezimalzahl		
actual_value	Dezimalzahl		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
measured_at	Zeitstempel		
higher_is_better	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.process_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
process_ci_id	UUID	•	
kind	Ganzzahl	•	
target_ci_id	UUID	•	
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.process_raci

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
process_ci_id	UUID	•	
raci_type	Ganzzahl	•	
user_id	UUID		
role_name	Text		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	
org_unit_ci_id	UUID		
position_id	UUID		

cmdb.process_step_annotations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
definition_id	UUID	•	
element_id	Text	•	
annotation_type	Ganzzahl	•	
title	Text	•	
description	Text		
nature	Ganzzahl		
mechanism	Ganzzahl		
frequency	Text		
control_owner	Text		
evidence	Text		
defense_line	Ganzzahl		
security_requirement_ref	Text		
risk_ref	UUID		
ci_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.product_responsibilities

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
catalog_product_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
org_unit_ci_id	UUID	•	
is_suggestion	Ja/Nein	•	
valid_from	Datum	•	
valid_to	Datum		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.rack_mounts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
rack_ci_id	UUID	•	
device_ci_id	UUID	•	
start_unit	Ganzzahl	•	
height_units	Ganzzahl	•	
side	Ganzzahl	•	
orientation	Text		
imported_from	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
is_full_depth	Ja/Nein	•	

cmdb.rack_reservations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
rack_ci_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
start_unit	Ganzzahl	•	
height_units	Ganzzahl	•	
side	Ganzzahl	•	
status	Ganzzahl	•	
description	Text	•	
requested_by	UUID	•	
valid_until	Datum		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.rack_types

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
manufacturer	Text	•	
model	Text	•	
slug	Text	•	
form_factor	Ganzzahl	•	
width_inches	Ganzzahl	•	
height_units	Ganzzahl	•	
starting_unit	Ganzzahl	•	
descending_units	Ja/Nein	•	
outer_width	Ganzzahl		
outer_height	Ganzzahl		
outer_depth	Ganzzahl		
outer_unit	Ganzzahl		
mounting_depth_mm	Ganzzahl		
max_weight	Dezimalzahl		
max_weight_unit	Ganzzahl		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
max_weight_grams	Ganzzahl		
comments	Text		
origin	Ganzzahl	•	
catalog_version	Text		
source_key	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.reconciliation_candidates

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scan_job_id	UUID	•	
ip_address	Text	•	
hostname	Text		
mac_address	Text		
os_name	Text		
os_version	Text		
match_type	Ganzzahl	•	
matched_ci_id	UUID		
status	Ganzzahl	•	
discovered_at	Zeitstempel	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

cmdb.state_law_references

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
family_key	Text	•	
family_label	Text	•	
state	Ganzzahl	•	
official_name	Text	•	
abbreviation	Text	•	
source_url	Text		
note	Text		
is_verified	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

Schema contracts

contracts.contract_ci_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
contract_id	UUID	•	
ci_id	UUID	•	
tenant_id	UUID	•	
linked_at	Zeitstempel	•	

contracts.contract_obligations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
contract_id	UUID	•	
title	Text	•	
description	Text		
due_date	Datum		
linked_ticket_id	UUID		
completed_at	Zeitstempel		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

contracts.contracts

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`title`	Text	•	Vertragsbezeichnung.
`vendor_name`	Text	•	Vertragspartner.
`contract_number`	Text		Vertragsnummer.
`type`	Text	•	Vertragsart.
`status`	Text	•	Vertragsstatus.
`start_date`	Datum		Vertragsbeginn.
`end_date`	Datum		Vertragsende.
`notice_period_days`	Ganzzahl	•	Kündigungsfrist in Tagen.
`annual_cost`	Dezimalzahl		Jährliche Kosten.
`payment_interval`	Text		
`cost_center`	Text		Kostenstelle.
`internal_owner_id`	UUID		
`external_contact`	Text		
`notes`	Text		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	
`cancellation_reminder_sent_on`	Datum		
`cancellation_effective_date`	Datum		Datum, zu dem die Kündigung wirkt.
`vendor_id`	UUID		

contracts.documents

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
contract_id	UUID	•	
file_name	Text	•	
file_size_bytes	Ganzzahl	•	
storage_path	Text	•	
description	Text		
version	Ganzzahl	•	
uploaded_by	UUID	•	
uploaded_at	Zeitstempel	•	

contracts.vendors

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
website	Text		
email	Text		
phone	Text		
notes	Text		
rating	Ganzzahl		

Schema datenschutz

datenschutz.avvs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processor_name	Text	•	
vendor_ci_id	UUID		
document_reference	Text		
valid_from	Zeitstempel		
valid_until	Zeitstempel		
note	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

datenschutz.data_breaches

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`reference`	Text	•	Referenz/Aktenzeichen.
`title`	Text	•	Kurzbeschreibung.
`description`	Text	•	
`occurred_at`	Zeitstempel		
`became_known_at`	Zeitstempel	•	Zeitpunkt der Kenntnis.
`reported_to_authority_at`	Zeitstempel		Meldung an die Aufsichtsbehörde.
`subjects_notified_at`	Zeitstempel		
`status`	Text	•	Bearbeitungsstatus.
`risk`	Text	•	Risikoeinstufung.
`affected_subject_count`	Ganzzahl		Anzahl betroffener Personen.
`likely_consequences`	Text		
`measures`	Text		
`no_report_rationale`	Text		
`supervisory_authority`	Text		
`ticket_id`	UUID		
`processing_activity_id`	UUID		
`reporter_user_id`	UUID		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

datenschutz.data_categories

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`description`	Text		
`kind`	Text	•	
`special_permission`	Text		
`is_system_entry`	Ja/Nein	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

datenschutz.data_subject_categories

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`description`	Text		
`is_vulnerable`	Ja/Nein	•	
`is_system_entry`	Ja/Nein	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

datenschutz.dpia_assessments

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`processing_activity_id`	UUID	•	
`profiling`	Ja/Nein	•	
`large_scale_special_data`	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`systematic_monitoring`	Ja/Nein	•	
`new_technology`	Ja/Nein	•	
`data_matching`	Ja/Nein	•	
`vulnerable_subjects`	Ja/Nein	•	
`likelihood`	Ganzzahl	•	
`severity`	Ganzzahl	•	
`notes`	Text		
`assessed_at`	Zeitstempel		
`assessed_by`	UUID		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

datenschutz.processing_activities

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	Name der Verarbeitung.
`purpose`	Text	•	Zweck der Verarbeitung.
`description`	Text		
`department`	Text		Verantwortliche Abteilung.
`config_profile`	Text	•	
`status`	Text	•	Status des VVT-Eintrags.
`process_id`	UUID		
`location_id`	UUID		
`owner_user_id`	UUID		
`dsb_user_id`	UUID		
`tech_contact_user_id`	UUID		
`review_interval_months`	Ganzzahl	•	Überprüfungsintervall in Monaten.
`next_review_due_at`	Zeitstempel		Fälligkeit der nächsten Überprüfung.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
approved_at	Zeitstempel		Zeitpunkt der Freigabe.
approved_by	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

datenschutz.processing_approval_steps

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
sequence	Ganzzahl	•	
role	Text	•	
decision	Text	•	
decided_by	UUID		
decided_by_name	Text		
decided_at	Zeitstempel		
comment	Text		
signature	Text		

datenschutz.processing_audit_entries

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
action	Text	•	
field_name	Text		
old_value	Text		
new_value	Text		
user_id	UUID	•	
user_name	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
occurred_at	Zeitstempel	•	

datenschutz.processing_ci_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
ci_id	UUID	•	
role	Text		
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	

datenschutz.processing_data_mappings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
data_subject_category_id	UUID	•	
data_category_id	UUID	•	
note	Text		

datenschutz.processing_legal_bases

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
article6	Text	•	
purpose_scope	Text		
balancing_test	Text		
special_law	Text	•	
special_law_reference	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

datenschutz.processing_recipients

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
category	Text	•	
name	Text		
kind	Text	•	
country_code	Text	•	
safeguard	Text	•	
safeguard_reference	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

datenschutz.processing_tom_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
measure_id	UUID	•	
note	Text		

datenschutz.retention_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
processing_activity_id	UUID	•	
data_category_id	UUID		
retention_months	Ganzzahl	•	Aufbewahrungsdauer in Monaten.
trigger	Text	•	Fristbeginn-Auslöser.
action	Text	•	Aktion nach Fristablauf (z. B. Löschen).

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
legal_reference	Text		Gesetzliche Grundlage.
note	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

datenschutz.technical_organizational_measures

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
category	Text	•	
is_system_entry	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

Schema discovery

discovery.action1_connections

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
region	Ganzzahl	•	
is_active	Ja/Nein	•	
is_deleted	Ja/Nein	•	
client_id	Text	•	
encrypted_secret	Text		
secret_iv	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
organization_id	Text		
organization_name	Text		
sync_interval_minutes	Ganzzahl	•	
max_requests_per_minute	Ganzzahl	•	
personal_data_sources_enabled	Ja/Nein	•	
personal_data_changed_by	UUID		
personal_data_changed_at	Zeitstempel		
last_sync_at	Zeitstempel		
last_ok_at	Zeitstempel		
last_error	Text		
consecutive_failures	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

discovery.action1_data_sources

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
connection_id	UUID	•	
source_key	Text	•	
is_enabled	Ja/Nein	•	
changed_by	UUID		
changed_at	Zeitstempel	•	

discovery.action1_endpoint_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
connection_id	UUID	•	
endpoint_id	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
endpoint_name	Text		
serial	Text		
mac_address	Text		
ci_id	UUID		
inventory_host_id	UUID		
match_kind	Text		
first_seen_at	Zeitstempel	•	
last_seen_at	Zeitstempel	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

discovery.action1_query_jobs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
connection_id	UUID	•	
source_key	Text	•	
endpoint_id	Text		
state	Ganzzahl	•	
provider_job_id	Text		
attempts	Ganzzahl	•	
created_at	Zeitstempel	•	
next_check_at	Zeitstempel	•	
requested_at	Zeitstempel		
completed_at	Zeitstempel		
last_error	Text		
result_count	Ganzzahl	•	

discovery.attachment_contents

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
attachment_id	UUID	•	
data	Binär	•	

discovery.attachments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
entity_type	Text	•	
entity_id	UUID	•	
file_name	Text	•	
content_type	Text	•	
size_bytes	Ganzzahl	•	
storage_path	Text		
version	Ganzzahl	•	
previous_version_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
visibility	Ganzzahl	•	
journal_entry_id	UUID		

discovery.collector_inventory_applications

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
host_id	UUID	•	
name	Text	•	
publisher	Text		
version	Text		
sku	Text		
last_seen_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

discovery.collector_inventory_hosts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	●	
tenant_id	UUID	●	
collector_id	UUID	●	
ip_address	Text	●	
hostname	Text		
mac_address	Text		
os_name	Text		
os_version	Text		
facts_json	JSON	●	
first_seen_at	Zeitstempel	●	
last_seen_at	Zeitstempel	●	
review_status	Ganzzahl	●	
ci_id	UUID		
created_at	Zeitstempel	●	
created_by	UUID	●	
modified_at	Zeitstempel	●	
modified_by	UUID	●	

discovery.collector_report_batches

Spalte	Typ	Pflicht	Beschreibung
id	UUID	●	
tenant_id	UUID	●	
collector_id	UUID	●	
received_at	Zeitstempel	●	
receipt_json	Text	●	

discovery.deep_scan_jobs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	●	
tenant_id	UUID	●	
scan_type	Text	●	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
target	Text	•	
status	Text	•	
credential_id	Text		
started_at	Zeitstempel		
completed_at	Zeitstempel		
result_payload	Text		
error_message	Text		
assets_found	Ganzzahl	•	
affected_ci_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

discovery.discovery_collectors

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
site_name	Text		
network_ranges	Text	•	
status	Ganzzahl	•	
enrollment_token_hash	Text		
enrollment_expires_at	Zeitstempel		
enrolled_at	Zeitstempel		
credential_token_id	UUID		
machine_name	Text		
agent_version	Text		
last_seen_at	Zeitstempel		
last_scan_at	Zeitstempel		
last_scan_error	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
last_scan_host_count	Ganzzahl	•	
scan_interval_minutes	Ganzzahl	•	
scan_enabled	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

discovery.scan_jobs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
network	Text	•	
status	Text	•	
started_at	Zeitstempel		
completed_at	Zeitstempel		
devices_found	Ganzzahl	•	
error_message	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

discovery.scan_snapshots

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
scan_job_id	UUID	•	
snapshot_data	Text	•	
sequence_number	Ganzzahl	•	
captured_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Schema `drplanning`

drplanning.dr_plan_ci_links

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`dr_plan_id`	UUID	•	
`ci_id`	UUID	•	
`tenant_id`	UUID	•	

drplanning.dr_plans

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`scope`	Text		
`rto_hours`	Ganzzahl	•	
`rpo_hours`	Ganzzahl	•	
`tier`	Ganzzahl	•	
`owner_id`	UUID		
`escalation_chain`	Text		
`recovery_steps`	Text		
`rollback_steps`	Text		
`last_tested_at`	Datum		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

drplanning.dr_test_results

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`dr_plan_id`	UUID	•	
`tenant_id`	UUID	•	
`tester_id`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`test_type`	Text	•	
`test_date`	Datum	•	
`status`	Text	•	
`actual_rto_hours`	Dezimalzahl		
`actual_rpo_hours`	Dezimalzahl		
`findings`	Text		
`actions`	Text		
`created_at`	Zeitstempel	•	

Schema `identity`

identity.abac_rules

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`role_id`	UUID	•	
`permission_id`	UUID	•	
`description`	Text	•	
`expression_json`	JSON	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

identity.access_review_campaigns

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`description`	Text	•	
`status`	Text	•	
`start_date`	Zeitstempel	•	
`due_date`	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_at`	Zeitstempel	•	
`created_by_user_id`	UUID	•	
`completed_at`	Zeitstempel		

identity.access_review_items

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`campaign_id`	UUID	•	
`user_id`	UUID	•	
`role_id`	UUID	•	
`decision`	Text	•	
`reviewer_comment`	Text		
`reviewed_by_user_id`	UUID		
`reviewed_at`	Zeitstempel		

identity.approval_delegations

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`from_user_id`	UUID	•	
`to_user_id`	UUID	•	
`from_date`	Zeitstempel	•	
`to_date`	Zeitstempel	•	
`note`	Text		
`created_at`	Zeitstempel	•	
`revoked_at`	Zeitstempel		

identity.idp_configurations

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`type`	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
is_active	Ja/Nein	•	
is_default	Ja/Nein	•	
sort_order	Ganzzahl	•	
config_json	JSON	•	
attribute_mapping	JSON	•	
group_mapping	JSON	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.incompatible_role_pairs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
role_a_id	UUID	•	
role_b_id	UUID	•	
reason	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.ldap_sync_logs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
idp_configuration_id	UUID	•	
started_at	Zeitstempel	•	
finished_at	Zeitstempel		
users_created	Ganzzahl	•	
users_updated	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
users_deactivated	Ganzzahl	•	
status	Text	•	
error_message	Text		
users_without_role	Ganzzahl	•	
groups_matched	Ganzzahl	•	

identity.mfa_devices

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
user_id	UUID	•	
type	Ganzzahl	•	
friendly_name	Text		
secret_or_credential_id	Text	•	
is_verified	Ja/Nein	•	
last_used_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.mfa_policies

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
enforce_for_all	Ja/Nein	•	
enforce_for_roles	JSON	•	
excluded_cidr_ranges	JSON	•	
grace_period_hours	Ganzzahl	•	
require_for_admin_roles	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	

identity.mobile_devices

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
device_id	Text	•	
device_name	Text	•	
platform	Text	•	
app_version	Text		
push_token	Text		
push_token_updated_at	Zeitstempel		
first_seen_at	Zeitstempel	•	
last_seen_at	Zeitstempel	•	
revoked_at	Zeitstempel		
is_revoked	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.password_reset_tokens

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
token_hash	Text	•	
expires_at	Zeitstempel	•	
created_at	Zeitstempel	•	
used_at	Zeitstempel		

identity.permission_group_permissions

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
permission_group_id	UUID	•	
permission_id	UUID	•	

identity.permission_groups

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
name	Text	•	
description	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.permissions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
key	Text	•	
description	Text		
module	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.role_permission_groups

Spalte	Typ	Pflicht	Beschreibung
role_id	UUID	•	
permission_group_id	UUID	•	

identity.role_permissions

Spalte	Typ	Pflicht	Beschreibung
role_id	UUID	•	
permission_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

identity.roles

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
name	Text	•	Rollenname.
description	Text		
is_system_role	Ja/Nein	•	
is_active	Ja/Nein	•	
color	Text		
icon	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

identity.sso_sessions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
provider_id	UUID	•	
refresh_token_protected	Text		
id_token_protected	Text		
created_at	Zeitstempel	•	
last_renewed_at	Zeitstempel	•	
revoked_at	Zeitstempel		
is_revoked	Ja/Nein	•	

identity.tenant_headquarters

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
label	Text		
street	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
house_number	Text		
postal_code	Text	•	
city	Text	•	
country	Text	•	

identity.tenant_roles

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
role_id	UUID	•	
is_enabled	Ja/Nein	•	

identity.tenants

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
name	Text	•	
slug	Text	•	
is_active	Ja/Nein	•	
logo_url	Text		
accent_color	Text		
primary_contact_email	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
employee_count	Ganzzahl		
size	Ganzzahl	•	

identity.user_roles

Spalte	Typ	Pflicht	Beschreibung
user_id	UUID	•	
role_id	UUID	•	
tenant_id	UUID	•	
valid_from	Zeitstempel		Beginn der Gültigkeit.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`valid_until`	Zeitstempel		Ende der Gültigkeit.
`assigned_by`	UUID		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

identity.user_sessions

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`user_id`	UUID	•	
`tenant_id`	UUID	•	
`refresh_token_hash`	Text	•	
`device_id`	Text		
`device_name`	Text		
`ip_address`	Text		
`user_agent`	Text		
`expires_at`	Zeitstempel	•	
`revoked_at`	Zeitstempel		
`is_revoked`	Ja/Nein	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

identity.users

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`login`	Text	•	Anmeldename.
`display_name`	Text	•	Anzeigename.
`email`	Text	•	E-Mail-Adresse.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
password_hash	Text		
status	Ganzzahl	●	Kontostatus.
is_system_account	Ja/Nein	●	Technisches/Systemkonto.
mfa_enforced	Ja/Nein	●	Mehr-Faktor-Authentisierung erzwungen.
failed_login_attempts	Ganzzahl	●	
locked_until	Zeitstempel		Sperre bis (Fehlanmeldungen).
last_login_at	Zeitstempel		Zeitpunkt der letzten Anmeldung.
external_idp_provider	Text		
external_idp_subject	Text		
department	Text		Abteilung.
phone_number	Text		
created_at	Zeitstempel	●	Anlagezeitpunkt.
created_by	UUID	●	
modified_at	Zeitstempel	●	
modified_by	UUID	●	
preferred_language	Text	●	
appearance_prefs	Text		
first_name	Text		
last_name	Text		
job_title	Text		Position/Funktion.
directory_attributes	JSON		

identity.wallboard_tokens

Spalte	Typ	Pflicht	Beschreibung
id	UUID	●	
tenant_id	UUID	●	
user_id	UUID	●	
name	Text	●	
token_hash	Text	●	
created_at	Zeitstempel	●	
created_by	UUID	●	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`expires_at`	Zeitstempel	•	
`revoked_at`	Zeitstempel		
`last_used_at`	Zeitstempel		

Schema `ipam`

ipam.aggregates

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`vrf_id`	UUID		
`prefix`	Text	•	
`description`	Text		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

ipam.dhcp_reservations

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`prefix_id`	UUID	•	
`ip_address`	Text	•	
`mac_address`	Text	•	
`hostname`	Text		
`description`	Text		
`assigned_ci_id`	UUID		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

ipam.ip_addresses

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
vrf_id	UUID		
prefix_id	UUID		
address	Text	•	IP-Adresse.
status	Ganzzahl	•	Vergabestatus.
dns_name	Text		Hinterlegter DNS-Name.
description	Text		Freitext.
assigned_ci_id	UUID		
nat_inside_id	UUID		
is_ephemeral	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
is_unconfirmed	Ja/Nein	•	Scan-Fund, noch nicht bestätigt.
last_seen_at	Zeitstempel		Letzter Scan-Kontakt.
scanned_hostname	Text		Beim Scan ermittelter Hostname.
scanned_mac_address	Text		Beim Scan ermittelte MAC.
assigned_interface_id	UUID		
scanned_properties	JSON		

ipam.ipam_settings

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
mark_dhcp_ranges	Ja/Nein	•	
modified_at	Zeitstempel	•	

ipam.prefixes

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
vrf_id	UUID		
aggregate_id	UUID		
network	Text	•	Zugehöriges Netz (Prefix).
status	Ganzzahl	•	Status des Netzes.
description	Text		Freitext.
site_ci_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
auto_scan_on_create	Ja/Nein	•	
dhcp_enabled	Ja/Nein	•	DHCP für dieses Netz aktiv.
dhcp_lease_duration	Text		
dhcp_options_json	Text		
dhcp_range_end	Text		Letzte DHCP-Adresse.
dhcp_range_start	Text		Erste DHCP-Adresse.
dhcp_server_id	UUID		
last_scan_at	Zeitstempel		Zeitpunkt des letzten Scans.
last_scan_found_count	Ganzzahl	•	Trefferzahl des letzten Scans.
last_scan_status	Text		
assigned_ci_id	UUID		
assigned_interface_id	UUID		

ipam.scan_ingest_batches

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
received_at	Zeitstempel	•	
agent_product	Text		
agent_name	Text		
integration_token_id	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`receipt_json`	Text	•	

ipam.vrfs

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`route_distinguisher`	Text		
`description`	Text		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

Schema `jobs`

jobs.mobile_sync_action

Spalte	Typ	Pflicht	Beschreibung
`tenant_id`	UUID	•	
`client_action_id`	UUID	•	
`kind`	Text	•	
`result`	Text	•	
`result_ref`	Text		
`error`	Text		
`processed_at`	Zeitstempel	•	

jobs.scheduled_job_config

Spalte	Typ	Pflicht	Beschreibung
`key`	Text	•	
`enabled`	Ja/Nein	•	
`schedule_kind`	Text	•	
`interval_minutes`	Ganzzahl	•	
`time_of_day_utc`	time without time zone	•	
`last_run_utc`	Zeitstempel		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`next_run_utc`	Zeitstempel		
`last_status`	Text		
`last_error`	Text		
`last_duration_ms`	Ganzzahl		

jobs.system_state

Spalte	Typ	Pflicht	Beschreibung
`key`	Text	•	
`value`	Text		
`updated_at`	Zeitstempel	•	

Schema `licensing`

licensing.license_assignments

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`license_entitlement_id`	UUID	•	
`tenant_id`	UUID	•	
`ci_id`	UUID		
`user_id`	UUID		
`assignment_type`	Text	•	
`assigned_at`	Zeitstempel	•	
`revoked_at`	Zeitstempel		

licensing.license_entitlements

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`product_name`	Text	•	Produktname.
`publisher`	Text	•	Softwarehersteller.
`version`	Text		
`license_type`	Text	•	Art der Lizenz.
`status`	Text	•	Lizenzstatus.
`purchased_quantity`	Ganzzahl	•	Erworbene Anzahl.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
cost_per_unit	Dezimalzahl		Kosten je Einheit.
contract_reference	Text		Referenz auf den Vertrag.
purchase_date	Datum		
expires_at	Datum		Ablaufdatum der Lizenz.
support_ends_at	Datum		Ende des Herstellersupports.
notes	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
contract_id	UUID		
support_reminder_sent_on	Datum		

licensing.license_pool_assignments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
pool_id	UUID	•	
entitlement_id	UUID	•	
ci_id	UUID		
user_id	UUID		
assigned_at	Zeitstempel	•	
revoked_at	Zeitstempel		

licensing.license_pools

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
cost_center	Text		
max_seats	Ganzzahl		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_at`	Zeitstempel	•	
`entitlement_id`	UUID		

Schema `notifications`

notifications.mail_accounts

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`purpose`	Ganzzahl	•	
`is_active`	Ja/Nein	•	
`is_deleted`	Ja/Nein	•	
`protocol`	Ganzzahl	•	
`host`	Text		
`port`	Ganzzahl	•	
`security`	Ganzzahl	•	
`tls_cert_policy`	Ganzzahl	•	
`custom_ca_thumbprint`	Text		
`auth_method`	Ganzzahl	•	
`username`	Text		
`encrypted_secret`	Text		
`secret_iv`	Text		
`oauth_provider`	Ganzzahl	•	
`encrypted_oauth_token`	Text		
`oauth_token_iv`	Text		
`oauth_expires_at`	Zeitstempel		
`from_address`	Text		
`display_name`	Text		
`reply_to_override`	Text		
`signature`	Text		
`support_group_id`	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
default_ticket_type	Text		
default_priority	Text		
inbox_folder	Text		
processed_folder	Text		
error_folder	Text		
pop3_delete_after_fetch	Ja/Nein	•	
poll_interval_seconds	Ganzzahl	•	
max_send_per_hour	Ganzzahl	•	
last_poll_at	Zeitstempel		
last_ok_at	Zeitstempel		
last_error	Text		
consecutive_failures	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
is_system_sender	Ja/Nein	•	
allowed_categories1	Text		
default_category1	Text		
default_category2	Text		

notifications.mail_outbox

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
mail_account_id	UUID		
support_group_id	UUID		
to_address	Text	•	
subject	Text	•	
html_body	Text	•	
status	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
attempts	Ganzzahl	•	
created_at	Zeitstempel	•	
next_attempt_at	Zeitstempel	•	
sent_at	Zeitstempel		
last_error	Text		
attachments_json	Text		
is_human_originated	Ja/Nein	•	
reply_to	Text		

notifications.messenger_accounts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
provider	Ganzzahl	•	
is_active	Ja/Nein	•	
is_deleted	Ja/Nein	•	
base_url	Text	•	
bot_display_name	Text		
encrypted_secret	Text		
secret_iv	Text		
body_template	Text		
default_channel	Text		
max_send_per_hour	Ganzzahl	•	
last_ok_at	Zeitstempel		
last_error	Text		
consecutive_failures	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

notifications.messenger_channel_targets

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
account_id	UUID	•	
event_type	Text	•	
channel_ref	Text	•	
is_active	Ja/Nein	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

notifications.messenger_outbox

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
account_id	UUID	•	
target_kind	Ganzzahl	•	
target_ref	Text	•	
recipient_user_id	UUID		
subject	Text	•	
body	Text	•	
event_type	Text		
reference	Text	•	
dedup_key	Text		
status	Ganzzahl	•	
send_state	Ganzzahl	•	
provider_code	Text		
attempts	Ganzzahl	•	
created_at	Zeitstempel	•	
next_attempt_at	Zeitstempel	•	
sent_at	Zeitstempel		
last_error	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

notifications.messenger_recipients

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
account_id	UUID	•	
user_id	UUID	•	
email	Text	•	
provider_user_id	Text		
resolved_at	Zeitstempel	•	
last_error	Text		

notifications.notification_templates

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID		
event_type	Text	•	
channel	Ganzzahl	•	
subject_template	Text	•	
body_template	Text	•	
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	

notifications.notifications

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
recipient_user_id	UUID	•	
subject	Text	•	
body	Text	•	
channel	Ganzzahl	•	
status	Ganzzahl	•	
event_type	Text		
related_entity_id	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
related_entity_type	Text		
created_at	Zeitstempel	•	
sent_at	Zeitstempel		
read_at	Zeitstempel		
error	Text		

notifications.outbox_messages

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
type	Text	•	
payload	JSON	•	
occurred_on	Zeitstempel	•	
processed_on	Zeitstempel		
error	Text		
tenant_id	UUID	•	

notifications.push_outbox

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
recipient_user_id	UUID	•	
device_record_id	UUID	•	
device_token	Text	•	
title	Text	•	
body	Text	•	
deep_link	Text	•	
event_type	Text	•	
status	Ganzzahl	•	
send_state	Ganzzahl	•	
provider_code	Text		
attempts	Ganzzahl	•	
created_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
next_attempt_at	Zeitstempel	•	
sent_at	Zeitstempel		
last_error	Text		

notifications.system_announcements

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
title	Text	•	
type	Text	•	
body	Text	•	
published_at	Zeitstempel	•	
scheduled_start	Zeitstempel		
scheduled_end	Zeitstempel		
is_active	Ja/Nein	•	

notifications.user_notification_preferences

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
event_type	Text	•	
channel	Ganzzahl	•	
is_enabled	Ja/Nein	•	

notifications.weather_warning_seen

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
warning_id	Text	•	
first_seen_at	Zeitstempel	•	
expires_at	Zeitstempel		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Schema `public`

public.__EFMigrationsHistory

Spalte	Typ	Pflicht	Beschreibung
`MigrationId`	Text	•	
`ProductVersion`	Text	•	

Schema `registry`

registry.cross_tenant_grants

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`grantor_tenant_id`	UUID	•	
`grantee_user_id`	UUID	•	
`scope_permissions`	Text	•	
`valid_from`	Zeitstempel	•	
`valid_until`	Zeitstempel		
`revoked`	Ja/Nein	•	
`created_by`	UUID	•	
`created_at`	Zeitstempel	•	

registry.email_signatures

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`support_group_id`	UUID		
`body_html`	Text	•	
`abbreviate_last_name`	Ja/Nein	•	
`enabled`	Ja/Nein	•	
`modified_at`	Zeitstempel	•	

registry.global_users

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`login`	Text	•	
`display_name`	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
email	Text	•	
password_hash	Text		
status	Ganzzahl	•	
mfa_enforced	Ja/Nein	•	
failed_login_attempts	Ganzzahl	•	
locked_until	Zeitstempel		
last_login_at	Zeitstempel		
created_at	Zeitstempel	•	
modified_at	Zeitstempel	•	
external_idp_provider	Text		
external_idp_subject	Text		

registry.integration_tokens

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
token_hash	Text	•	
scopes	Text	•	
created_by_user_id	UUID	•	
created_at	Zeitstempel	•	
last_used_at	Zeitstempel		
expires_at	Zeitstempel		
revoked_at	Zeitstempel		

registry.tenant_branding

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
company_name	Text	•	
logo	Binär		
logo_content_type	Text		
modified_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
city	Text		
country	Text		
legal_form	Text		
managing_directors	Text		
postal_code	Text		
register_court	Text		
register_number	Text		
street	Text		
vat_id	Text		
report_footer_text	Text		
report_header_text	Text		

registry.tenant_memberships

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
user_id	UUID	•	
tenant_id	UUID	•	
can_switch	Ja/Nein	•	
is_default	Ja/Nein	•	
created_at	Zeitstempel	•	

registry.tenants

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
slug	Text	•	
name	Text	•	
connection_string	Text	•	
status	Ganzzahl	•	
parent_tenant_id	UUID		
created_at	Zeitstempel	•	
modified_at	Zeitstempel	•	
quarantine_reason	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
quarantined_at	Zeitstempel		

Schema reporting

reporting.dashboard_layouts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
dashboard_type	Ganzzahl	•	
columns	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

reporting.dashboard_widgets

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
dashboard_layout_id	UUID	•	
type	Ganzzahl	•	
title	Text	•	
column	Ganzzahl	•	
row	Ganzzahl	•	
col_span	Ganzzahl	•	
row_span	Ganzzahl	•	
is_visible	Ja/Nein	•	
config_json	JSON	•	

reporting.report_definitions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
description	Text	•	
module	Text	•	
query_json	JSON	•	
is_system	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by_user_id	UUID	•	

reporting.report_histories

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
report_definition_id	UUID	•	
report_name	Text	•	
format	Text	•	
row_count	Ganzzahl	•	
file_size_bytes	Ganzzahl	•	
generated_at	Zeitstempel	•	
generated_by_user_id	UUID	•	
was_scheduled	Ja/Nein	•	

reporting.report_schedules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
report_definition_id	UUID	•	
cron_expression	Text	•	
recipient_emails	Liste	•	
format	Text	•	
is_active	Ja/Nein	•	
last_run_at	Zeitstempel		
next_run_at	Zeitstempel		
created_by_user_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Schema `security`

security.awareness_assignments

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`user_id`	UUID	•	
`title`	Text	•	
`description`	Text		
`kind`	Text	•	
`assigned_at`	Zeitstempel	•	
`due_at`	Zeitstempel		
`acknowledged_at`	Zeitstempel		

security.elementary_threats

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`catalog_id`	UUID	•	
`tenant_id`	UUID	•	
`external_id`	Text	•	
`title`	Text	•	
`description`	Text		
`is_custom`	Ja/Nein	•	

security.isms_audit_findings

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`audit_id`	UUID	•	
`finding_type`	Ganzzahl	•	
`clause_or_control`	Text		
`description`	Text	•	
`status`	Ganzzahl	•	
`nonconformity_id`	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

security.isms_documents

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`document_type`	Ganzzahl	•	
`version`	Text	•	
`owner`	Text		
`status`	Ganzzahl	•	
`approved_by`	UUID		
`approved_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

security.isms_management_reviews

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`review_date`	Zeitstempel	•	
`inputs`	Text	•	
`outputs`	Text		
`approved_by`	UUID		
`approved_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.isms_nonconformities

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
description	Text	•	Beschreibung der Abweichung.
source	Text		Herkunft (z. B. Audit).
root_cause	Text		
corrective_action	Text		Geplante Korrektur.
due_date	Zeitstempel		Fälligkeit der Korrekturmaßnahme.
responsible_user_id	UUID		
effectiveness_check	Text		
effectiveness_verified_at	Zeitstempel		
status	Ganzzahl	•	Bearbeitungsstand.
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.isms_objectives

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
title	Text	•	
description	Text		
metric	Text		
target_value	Text		
current_value	Text		
due_date	Zeitstempel		
status	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

security.security_audits

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`scope_id`	UUID	•	
`name`	Text	•	
`status`	Text	•	
`started_at`	Zeitstempel	•	
`closed_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

security.security_baustein_assignments

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`target_object_id`	UUID	•	
`catalog_item_id`	UUID	•	
`baustein_external_id`	Text	•	
`baustein_title`	Text	•	
`layer`	Text		
`source`	Text	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	

security.security_catalog_items

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
catalog_id	UUID	•	
tenant_id	UUID	•	
external_id	Text	•	
title	Text	•	
layer	Text	•	
description	Text		
applicable_object_types	Liste		
is_custom	Ja/Nein	•	

security.security_catalogs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
framework	Text	•	
version	Text	•	
oscal_source	Text		
imported_at	Zeitstempel	•	
is_active	Ja/Nein	•	
raw_oscal_json	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
source_hash	Text		
available_update_hash	Text		
available_update_version	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

security.security_check_results

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
audit_id	UUID	•	
target_object_id	UUID	•	
baustein_external_id	Text	•	
baustein_title	Text	•	
requirement_external_id	Text	•	
requirement_title	Text	•	
level	Text	•	
status	Text	•	
rationale	Text		
responsible_user_id	UUID		
due_date	Datum		
measure_status	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.security_component_definitions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_key	Text	•	
title	Text	•	
version	Text	•	
oscal_version	Text		
source_url	Text		
remarks	Text		
is_draft	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
imported_at	Zeitstempel	•	
resolved_count	Ganzzahl	•	
unresolved_count	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.security_component_implementations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
definition_id	UUID	•	
tenant_id	UUID	•	
component_title	Text	•	
component_type	Text		
component_description	Text		
control_id_ref	Text	•	
requirement_external_id	Text		
description	Text		
reference_links	Liste		
adopted_at	Zeitstempel		
adopted_by	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.security_control_mappings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_key	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
collection_title	Text	•	
collection_version	Text	•	
method	Text		
source_catalog	Text		
target_catalog	Text		
source_control_id	Text	•	
target_control_id	Text	•	
relationship	Text	•	
is_resolved	Ja/Nein	•	
imported_at	Zeitstempel	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.security_profile_exclusions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
profile_id	UUID	•	
requirement_external_id	Text	•	
justification	Text	•	
created_at	Zeitstempel	•	

security.security_profiles

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
include_basis	Ja/Nein	•	
include_standard	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
include_erhoeht	Ja/Nein	•	
is_blueprint	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.security_requirements

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
catalog_item_id	UUID	•	
tenant_id	UUID	•	
external_id	Text	•	
title	Text	•	
description	Text		
level	Text	•	
is_obsolete	Ja/Nein	•	
cross_references	Liste		
is_custom	Ja/Nein	•	
alt_identifier	Text		

security.security_risk_analyses

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
name	Text	•	
description	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

security.security_risks

Spalte	Typ	Pflicht	Beschreibung
id	UUID	●	
tenant_id	UUID	●	
analysis_id	UUID	●	
target_object_id	UUID		
threat_external_id	Text		
threat_title	Text		Zugrunde liegende Bedrohung.
title	Text	●	Risikobezeichnung.
description	Text		
likelihood	Ganzzahl	●	Eintrittswahrscheinlichkeit.
impact	Ganzzahl	●	Schadenshöhe.
score	Ganzzahl	●	Bruttorisiko (Wahrscheinlichkeit × Auswirkung).
category	Text	●	Risikokategorie (z. B. Hoch).
strategy	Text	●	Behandlungsstrategie.
residual_likelihood	Ganzzahl	●	
residual_impact	Ganzzahl	●	
residual_score	Ganzzahl	●	Risiko nach Maßnahmen.
residual_category	Text	●	
status	Text	●	Bearbeitungsstatus.
responsible_user_id	UUID		
accepted_by	UUID		
accepted_at	Zeitstempel		
acceptance_rationale	Text		
created_at	Zeitstempel	●	
created_by	UUID	●	
modified_at	Zeitstempel	●	
modified_by	UUID	●	
treatment_controls	Liste	●	

security.security_scopes

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
scope_statement	Text		
interested_parties	Text		
isms_policy	Text		

security.security_target_objects

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
scope_id	UUID	•	
ci_id	UUID		
manual_name	Text		
object_type	Text	•	
confidentiality	Text		
integrity	Text		
availability	Text		
confidentiality_rationale	Text		
integrity_rationale	Text		
availability_rationale	Text		
assessed_at	Zeitstempel		
assessed_by	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	

security.soa_entries

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
catalog_item_id	UUID	•	
control_external_id	Text	•	Kontroll-Kennung (z. B. A.5.1).
control_title	Text	•	Titel der Kontrolle.
is_applicable	Ja/Nein	•	Kontrolle ist anwendbar.
justification	Text		Begründung zu Anwendbarkeit/Ausschluss.
implementation_status	Ganzzahl	•	Soll-Ist-Umsetzungsstatus.
responsible_party	Text		Verantwortliche Stelle.
implementation_reference	Text		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

security.vulnerability_findings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ci_id	UUID	•	
advisory_key	Text	•	
product_cpe	Text	•	
installed_version	Text	•	
ci_name	Text	•	
product_name	Text	•	
publisher	Text		
advisory_title	Text	•	
cve_ids	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`cvss_base_score`	Dezimalzahl		
`cvss_vector`	Text		
`known_exploited`	Ja/Nein	•	
`match_kind`	Ganzzahl	•	
`matched_range`	Text		
`mapping_rule_id`	UUID		
`status`	Ganzzahl	•	
`triage`	Ganzzahl	•	
`triage_reason`	Text		
`accepted_until`	Datum		
`triaged_by`	UUID		
`triaged_at`	Zeitstempel		
`security_risk_id`	UUID		
`change_request_id`	UUID		
`first_seen_at`	Zeitstempel	•	
`last_seen_at`	Zeitstempel	•	
`resolved_at`	Zeitstempel		

Schema `selforg`

selforg.calendar_feed_tokens

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`user_id`	UUID	•	
`token_hash`	Text	•	
`revoked_at`	Zeitstempel		
`is_revoked`	Ja/Nein	•	
`last_used_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_by	UUID	•	

selforg.exchange_connections

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ews_url	Text		
autodiscover_domain	Text		
exchange_version	Text	•	
auth_mode	Text	•	
username	Text		
secret_protected	Text		
tls_policy	Text	•	
custom_ca_thumbprint	Text		
read_enabled	Ja/Nein	•	
write_enabled	Ja/Nein	•	
read_mode	Text	•	
last_ok_at	Zeitstempel		
last_error_at	Zeitstempel		
last_error	Text		
consecutive_failures	Ganzzahl	•	
suspended_until	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
allow_user_delegation	Ja/Nein	•	

selforg.mailbox_bindings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`user_id`	UUID	•	
`smtp_address`	Text	•	
`is_active`	Ja/Nein	•	
`enabled_at`	Zeitstempel	•	
`revoked_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

selforg.personal_task_links

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`task_id`	UUID	•	
`object_type`	Text	•	
`object_id`	UUID	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

selforg.personal_tasks

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`owner_user_id`	UUID	•	
`board_id`	UUID	•	
`column_id`	UUID	•	
`title`	Text	•	
`notes`	Text		
`priority`	Text	•	
`due_at`	Zeitstempel		
`planned_for`	Datum		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
estimate_minutes	Ganzzahl		
sort_order	Ganzzahl	•	
completed_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
reminded_up_to	Zeitstempel		

selforg.reminder_settings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
lead_minutes	Ganzzahl	•	
enabled	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

selforg.task_board_columns

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
board_id	UUID	•	
name	Text	•	
sort_order	Ganzzahl	•	
semantic_state	Text	•	
is_default_for_state	Ja/Nein	•	
wip_limit	Ganzzahl		

selforg.task_boards

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`owner_type`	Text	•	
`owner_id`	UUID	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

selforg.task_handovers

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`task_id`	UUID	•	
`from_user_id`	UUID	•	
`to_user_id`	UUID	•	
`status`	Text	•	
`message`	Text		
`decision_note`	Text		
`requested_at`	Zeitstempel	•	
`decided_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

Schema `ticketing`

ticketing.approval_tiers

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`name`	Text	•	
`min_amount`	Dezimalzahl	•	
`approver_id`	UUID	•	
`is_active`	Ja/Nein	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

ticketing.calendars

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`time_zone`	Text	•	
`is24x7`	Ja/Nein	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

ticketing.canned_responses

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`title`	Text	•	
`body`	Text	•	
`category`	Text		
`is_shared`	Ja/Nein	•	
`creator_id`	UUID	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
modified_at	Zeitstempel	•	
modified_by	UUID	•	
body_format	Ganzzahl	•	

ticketing.change_approval_settings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
allow_self_approval	Ja/Nein	•	
approver_user_ids	Liste	•	

ticketing.change_request_approvals

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
change_request_id	UUID	•	
approver_id	UUID	•	
is_approved	Ja/Nein	•	
comment	Text		
decided_at	Zeitstempel	•	

ticketing.change_request_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
change_request_id	UUID	•	
link_type	Ganzzahl	•	
target_id	UUID	•	
target_label	Text	•	
linked_at	Zeitstempel	•	

ticketing.change_requests

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
change_number	Text	•	
type	Text	•	
status	Text	•	
risk_level	Text	•	
title	Text	•	
description	Text		
justification	Text		
rollback_plan	Text		
test_results	Text		
requester_id	UUID	•	
assignee_id	UUID		
affected_ci_id	UUID		
planned_start_at	Zeitstempel		
planned_end_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.contacts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
email	Text	•	
display_name	Text	•	
organization	Text		
phone	Text		
notes	Text		
source	Ganzzahl	•	
is_active	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.csat_ratings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
ticket_id	UUID	•	
tenant_id	UUID	•	
stars	Ganzzahl	•	
comment	Text		
rated_at	Zeitstempel	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.defaults_seed_markers

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
seed_key	Text	•	
applied_at	Zeitstempel	•	

ticketing.holidays

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
calendar_id	UUID	•	
date	Datum	•	
name	Text	•	
is_half_day	Ja/Nein	•	
half_day_start_time	time without time zone		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ticketing.journal

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
ticket_id	UUID	•	
tenant_id	UUID	•	
entry_type	Ganzzahl	•	
body	Text	•	
is_private	Ja/Nein	•	
author_id	UUID	•	
created_at	Zeitstempel	•	
body_format	Ganzzahl	•	
direction	Ganzzahl	•	
event_code	Text		
event_args_json	JSON		
mail_recipients_json	JSON		

ticketing.knowledge_article_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_article_id	UUID	•	
target_article_id	UUID	•	
link_type	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	

ticketing.knowledge_article_status_history

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
article_id	UUID	•	
tenant_id	UUID	•	
from_status	Ganzzahl	•	
to_status	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
by_user_id	UUID	•	
at	Zeitstempel	•	
comment	Text		

ticketing.knowledge_article_translations

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
article_id	UUID	•	
language	Text	•	
title	Text	•	
symptom	Text		
resolution	Text	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.knowledge_article_versions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
article_id	UUID	•	
version_no	Ganzzahl	•	
title	Text	•	
symptom	Text		
resolution	Text	•	
category1	Text		
category2	Text		
keywords	Text		
visibility	Ganzzahl	•	
changed_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
changed_at	Zeitstempel	•	
change_note	Text		

ticketing.knowledge_article_views

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
article_id	UUID	•	
user_id	UUID	•	
viewed_at	Zeitstempel	•	

ticketing.knowledge_articles

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
title	Text	•	
symptom	Text		
resolution	Text	•	
category1	Text		
category2	Text		
keywords	Text		
status	Ganzzahl	•	
source_ticket_id	UUID		
view_count	Ganzzahl	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
visibility	Ganzzahl	•	
submitted_by	UUID		
submitted_at	Zeitstempel		
reviewed_by	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
reviewed_at	Zeitstempel		
tags	Liste	•	
helpful_count	Ganzzahl	•	
not_helpful_count	Ganzzahl	•	
reuse_count	Ganzzahl	•	
improvement_flagged	Ja/Nein	•	
improvement_note	Text		
improvement_flagged_by	UUID		
improvement_flagged_at	Zeitstempel		
deflection_count	Ganzzahl	•	
owner_id	UUID		
review_by_date	Zeitstempel		
is_stale	Ja/Nein	•	

ticketing.knowledge_feedback

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
article_id	UUID	•	
is_helpful	Ja/Nein	•	
comment	Text		
by_user_id	UUID	•	
created_at	Zeitstempel	•	

ticketing.knowledge_search_logs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
term	Text	•	
result_count	Ganzzahl	•	
user_id	UUID		
searched_at	Zeitstempel	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ticketing.knowledge_subscriptions

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
user_id	UUID	•	
article_id	UUID		
category1	Text		
created_at	Zeitstempel	•	

ticketing.maintenance_windows

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
title	Text	•	
description	Text		
starts_at	Zeitstempel	•	
ends_at	Zeitstempel	•	
scope_ci_id	UUID		
scope_category1	Text		
mute_notifications	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.monitoring_alerts

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_id	UUID	•	
correlation_key	Text	•	
host	Text	•	
service	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
severity	Text	•	
subject	Text	•	
state	Ganzzahl	•	
first_seen_at	Zeitstempel	•	
last_seen_at	Zeitstempel	•	
resolve_due_at	Zeitstempel		
resolved_at	Zeitstempel		
occurrence_count	Ganzzahl	•	
state_change_count	Ganzzahl	•	
is_flapping	Ja/Nein	•	
ticket_id	UUID		
ci_id	UUID		
suppression_reason	Text		
storm_parent_alert_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.monitoring_events

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_id	UUID	•	
alert_id	UUID		
external_event_id	Text		
correlation_key	Text	•	
host	Text	•	
service	Text		
severity	Text	•	
event_type	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
message	Text		
payload_json	Text		
occurred_at	Zeitstempel	•	
received_at	Zeitstempel	•	
outcome	Text	•	
reason	Text		

ticketing.monitoring_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
source_id	UUID		
name	Text	•	
is_active	Ja/Nein	•	
sort_order	Ganzzahl	•	
stop_processing	Ja/Nein	•	
host_pattern	Text		
service_pattern	Text		
severity	Text		
label	Text		
suppress_ticket	Ja/Nein	•	
category1	Text		
category2	Text		
support_group_id	UUID		
assignee_id	UUID		
ticket_type	Ganzzahl		
priority	Ganzzahl		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ticketing.monitoring_sources

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
kind	Text	•	
is_active	Ja/Nein	•	
credential_token_id	UUID		
default_support_group_id	UUID		
severity_map_json	Text		
default_priority	Ganzzahl	•	
resolve_delay_minutes	Ganzzahl	•	
storm_threshold_per_minute	Ganzzahl	•	
flapping_threshold	Ganzzahl	•	
flapping_window_minutes	Ganzzahl	•	
respect_human_work	Ja/Nein	•	
deep_link_template	Text		
last_event_at	Zeitstempel		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.number_counters

Spalte	Typ	Pflicht	Beschreibung
tenant_id	UUID	•	
key	Text	•	
value	Ganzzahl	•	

ticketing.priority_matrix

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
urgency	Ganzzahl	•	
impact	Ganzzahl	•	
priority	Ganzzahl	•	

ticketing.problem_incident_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
problem_id	UUID	•	
ticket_id	UUID	•	
linked_at	Zeitstempel	•	

ticketing.problem_journal

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
problem_id	UUID	•	
tenant_id	UUID	•	
entry_type	Text	•	
body	Text	•	
is_private	Ja/Nein	•	
author_id	UUID	•	
created_at	Zeitstempel	•	

ticketing.problems

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
problem_number	Text	•	
status	Text	•	
title	Text	•	
description	Text		
symptoms	Text		
root_cause	Text		
workaround	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
resolution	Text		
is_known_error	Ja/Nein	•	
reporter_id	UUID	•	
assignee_id	UUID		
affected_ci_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.routing_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
category1	Text	•	
category2	Text		
target_support_group	Text	•	
default_assignee_id	UUID		
sort_order	Ganzzahl	•	
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.service_articles

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
category_id	UUID	•	
name	Text	•	
description	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
default_priority	Ganzzahl	•	
default_assignee_group_id	UUID		
form_fields	JSON	•	
expected_hours	Ganzzahl	•	
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
requires_approval	Ja/Nein	•	
approver_id	UUID		
ticket_category1	Text		
ticket_category2	Text		

ticketing.service_categories

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
description	Text		
sort_order	Ganzzahl	•	
is_active	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.sla_escalation_settings

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
is_enabled	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
level1_after_minutes	Ganzzahl	•	
level2_after_minutes	Ganzzahl	•	
level3_after_minutes	Ganzzahl	•	

ticketing.sla_profiles

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
is_default	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
calendar_id	UUID		
is_active	Ja/Nein	•	

ticketing.sla_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
sla_profile_id	UUID	•	
priority	Ganzzahl	•	
reaction_time_minutes	Ganzzahl	•	
resolution_time_minutes	Ganzzahl	•	
escalation_l1_minutes	Ganzzahl		
escalation_l2_minutes	Ganzzahl		
escalation_mgmt_minutes	Ganzzahl		

ticketing.support_group_members

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
support_group_id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
member_type	Ganzzahl	•	
member_id	UUID	•	
created_at	Zeitstempel	•	
created_by	UUID	•	

ticketing.support_groups

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
is_active	Ja/Nein	•	
response_target_minutes	Ganzzahl	•	
resolution_target_minutes	Ganzzahl	•	
escalation_l1_recipient_id	UUID		
escalation_l2_recipient_id	UUID		
escalation_mgmt_recipient_id	UUID		
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.ticket_appointments

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ticket_id	UUID	•	
organizer_user_id	UUID	•	
attendee_email	Text	•	
attendee_name	Text		
subject	Text	•	
location	Text		
description	Text		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
starts_at	Zeitstempel	•	
ends_at	Zeitstempel	•	
status	Ganzzahl	•	
responded_at	Zeitstempel		
uid	Text	•	
response_token	UUID	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.ticket_automation_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
name	Text	•	
is_active	Ja/Nein	•	
sort_order	Ganzzahl	•	
match_type	Ganzzahl		
match_category1	Text		
subject_contains	Text		
reporter_email_domain	Text		
set_priority	Ganzzahl		
set_support_group	Text		
set_assignee_id	UUID		
add_watcher_email	Text		
stop_processing	Ja/Nein	•	
set_category1	Text		
set_category2	Text		
match_mail_account_id	UUID		

ticketing.ticket_call_logs

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ticket_id	UUID	•	
direction	Ganzzahl	•	
remote_number	Text	•	
notes	Text		
started_at	Zeitstempel	•	
duration_seconds	Ganzzahl	•	
logged_by	UUID	•	
logged_at	Zeitstempel	•	

ticketing.ticket_categories

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
level	Ganzzahl	•	
name	Text	•	
parent_name	Text		
sort_order	Ganzzahl	•	
is_active	Ja/Nein	•	
ci_class_filter	Text		

ticketing.ticket_ci_links

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ticket_id	UUID	•	
ci_id	UUID	•	
note	Text		
added_at	Zeitstempel	•	
added_by	UUID	•	

ticketing.ticket_knowledge_links

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`ticket_id`	UUID	•	
`article_id`	UUID	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

ticketing.ticket_links

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`source_ticket_id`	UUID	•	
`target_ticket_id`	UUID	•	
`link_type`	Text	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

ticketing.ticket_metric_snapshots

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`day`	Datum	•	Kennzahlen-Tag.
`captured_at`	Zeitstempel	•	
`created_count`	Ganzzahl	•	Am Tag erstellte Tickets.
`resolved_count`	Ganzzahl	•	Am Tag gelöste Tickets.
`open_backlog`	Ganzzahl	•	Offene Tickets am Tagesende.
`resolution_sla_total`	Ganzzahl	•	Bewertete Lösungs-SLAs.
`resolution_sla_met`	Ganzzahl	•	Innerhalb Lösungs-SLA.
`response_sla_total`	Ganzzahl	•	Bewertete Reaktions-SLAs.
`response_sla_met`	Ganzzahl	•	Innerhalb Reaktions-SLA.
`csat_count`	Ganzzahl	•	
`csat_sum`	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

ticketing.ticket_tasks

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`ticket_id`	UUID	•	
`title`	Text	•	
`assignee_id`	UUID		
`is_done`	Ja/Nein	•	
`sort_order`	Ganzzahl	•	
`completed_at`	Zeitstempel		
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	

ticketing.ticket_templates

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`name`	Text	•	
`type`	Text	•	
`default_priority`	Text	•	
`subject_template`	Text	•	
`description_template`	Text		
`is_active`	Ja/Nein	•	
`created_at`	Zeitstempel	•	
`created_by`	UUID	•	
`modified_at`	Zeitstempel	•	
`modified_by`	UUID	•	

ticketing.ticket_watchers

Spalte	Typ	Pflicht	Beschreibung
`id`	UUID	•	
`tenant_id`	UUID	•	
`ticket_id`	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
watcher_user_id	UUID		
watcher_email	Text		
display_name	Text		
added_at	Zeitstempel	•	
added_by	UUID	•	

ticketing.ticket_worklogs

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
ticket_id	UUID	•	
tenant_id	UUID	•	
agent_id	UUID	•	
minutes_spent	Ganzzahl	•	
note	Text		
work_date	Datum	•	
created_at	Zeitstempel	•	
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	

ticketing.tickets

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
tenant_id	UUID	•	
ticket_number	Text	•	Eindeutige Ticketnummer.
type	Ganzzahl	•	Ticket-Art.
status	Ganzzahl	•	Bearbeitungsstatus.
priority	Ganzzahl	•	Priorität.
subject	Text	•	Betreff/Kurzbeschreibung.
description	Text		
reporter_id	UUID	•	
assignee_id	UUID		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
support_group_id	UUID		
affected_ci_id	UUID		
sla_profile_id	UUID		
parent_ticket_id	UUID		
sla_due_at	Zeitstempel		Fällig laut Lösungs-SLA.
escalation_level	Ganzzahl	•	0 = keine Eskalation.
root_cause	Text		
workaround	Text		
is_known_error	Ja/Nein	•	Als bekannter Fehler markiert.
created_at	Zeitstempel	•	Erstellzeitpunkt.
created_by	UUID	•	
modified_at	Zeitstempel	•	
modified_by	UUID	•	
support_group_name	Text		Zugewiesenes Support-Team.
category1	Text		Hauptkategorie.
category2	Text		Unterkategorie.
effort_expected_minutes	Ganzzahl	•	
follow_up_at	Zeitstempel		
hold_reason	Text		
resolution_code	Text		
reopen_count	Ganzzahl	•	
reporter_email	Text		E-Mail des Melders.
sla_response_due_at	Zeitstempel		
first_responded_at	Zeitstempel		Zeitpunkt der Erstreaktion.
sla_response_breached_at	Zeitstempel		Zeitpunkt einer Reaktions-SLA-Verletzung.
sla_paused_at	Zeitstempel		
approval_status	Text	•	Genehmigungsstatus (z. B. Pending/Approved).
approver_id	UUID		
approval_decided_at	Zeitstempel		
approval_comment	Text		
estimated_cost	Dezimalzahl		

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
description_format	Ganzzahl	•	
merged_into_ticket_id	UUID		
decision_maker	Text		
reporter_language	Text	•	
resolved_at	Zeitstempel		
closed_at	Zeitstempel		
csat_token	UUID		
csat_requested_at	Zeitstempel		
last_reminder_at	Zeitstempel		
vendor_ci_id	UUID		
uc_reference	Text		
uc_due_at	Zeitstempel		
uc_breached_at	Zeitstempel		
file_reference	Text		
statutory_due_at	Zeitstempel		
statutory_breached_at	Zeitstempel		
channel	Ganzzahl	•	
eakte_reference	Text		
deferred_until	Zeitstempel		
follow_up_reminder_stage	Ganzzahl	•	
deferred_reminder_stage	Ganzzahl	•	
is_due_date_manual	Ja/Nein	•	
due_reminder_stage	Ganzzahl	•	
decision_maker_user_id	UUID		
decision_maker_email	Text		
sla_breached_at	Zeitstempel		
major_incident_declared_at	Zeitstempel		

ticketing.work_slots

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
calendar_id	UUID	•	
day_of_week	Ganzzahl	•	
start_time	time without time zone	•	
end_time	time without time zone	•	

Schema vulncatalog

vulncatalog.advisories

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
source	Ganzzahl	•	
source_document_id	Text	•	
title	Text	•	
cve_ids	Text		
summary	Text		
published_at	Zeitstempel		
last_modified_at	Zeitstempel		
cvss_base_score	Dezimalzahl		
cvss_vector	Text		
cvss_version	Text		
known_exploited	Ja/Nein	•	
signature_state	Ganzzahl	•	
remediations_json	JSON		
source_url	Text		
fetched_at	Zeitstempel	•	

vulncatalog.affected_products

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
advisory_id	UUID	•	
cpe_part	Text	•	
cpe_vendor	Text	•	
cpe_product	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
cpe_target_sw	Text	•	
exact_version	Text		
version_start_including	Text		
version_start_excluding	Text		
version_end_including	Text		
version_end_excluding	Text		
source_product_name	Text		

vulncatalog.cpe_mapping_rules

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
ordinal	Ganzzahl	•	
publisher_pattern	Text		
product_pattern	Text	•	
version_pattern	Text		
part	Text	•	
vendor	Text		
product	Text		
target_sw	Text		
suppress	Ja/Nein	•	
origin	Ganzzahl	•	
note	Text		
is_enabled	Ja/Nein	•	
created_at	Zeitstempel	•	
created_by	UUID		
modified_at	Zeitstempel	•	
modified_by	UUID		

vulncatalog.feeds

Spalte	Typ	Pflicht	Beschreibung
id	UUID	•	
kind	Ganzzahl	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`name`	Text	•	
`url`	Text	•	
`is_enabled`	Ja/Nein	•	
`secret_protected`	Text		
`proxy_url`	Text		
`last_run_at`	Zeitstempel		
`last_success_at`	Zeitstempel		
`last_error`	Text		
`last_error_at`	Zeitstempel		
`last_document_count`	Ganzzahl	•	
`incremental_cursor`	Zeitstempel		

Schema `workflow`

workflow.instances

Spalte	Typ	Pflicht	Beschreibung
`Id`	UUID	•	
`TenantId`	UUID	•	
`TemplateId`	UUID	•	
`TemplateName`	Text	•	
`TriggerContext`	JSON	•	
`Status`	Ganzzahl	•	
`CurrentStepOrder`	Ganzzahl	•	
`StartedByUserId`	UUID	•	
`StartedAt`	Zeitstempel	•	
`CompletedAt`	Zeitstempel		

workflow.step_definitions

Spalte	Typ	Pflicht	Beschreibung
`Id`	UUID	•	
`TemplateId`	UUID	•	
`Order`	Ganzzahl	•	
`Name`	Text	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`StepType`	Ganzzahl	•	
`AssigneeUserId`	UUID		
`AssigneeRoleId`	Text		
`TimeoutHours`	Ganzzahl		
`EscalationUserId`	UUID		
`Config`	JSON	•	

workflow.tasks

Spalte	Typ	Pflicht	Beschreibung
`Id`	UUID	•	
`TenantId`	UUID	•	
`InstanceId`	UUID	•	
`StepOrder`	Ganzzahl	•	
`StepName`	Text	•	
`StepType`	Ganzzahl	•	
`AssigneeUserId`	UUID		
`Status`	Ganzzahl	•	
`CreatedAt`	Zeitstempel	•	
`DueAt`	Zeitstempel		
`CompletedAt`	Zeitstempel		
`CompletedByUserId`	UUID		
`Notes`	Text		

workflow.templates

Spalte	Typ	Pflicht	Beschreibung
`Id`	UUID	•	
`TenantId`	UUID	•	
`Name`	Text	•	
`Description`	Text		
`TriggerType`	Ganzzahl	•	
`TriggerConfig`	JSON	•	
`IsActive`	Ja/Nein	•	

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Spalte	Typ	Pflicht	Beschreibung
`IsBuiltIn`	Ja/Nein	•	
`CreatedAt`	Zeitstempel	•	
`ModifiedAt`	Zeitstempel	•	

Verwandte Themen

- **ID-27a – Reports & SQL-Abfragen** – eigene Abfragen im Report-Baukasten
- **ID-27 – Modul Reporting & Dashboards** – fertige Berichte und Dashboards
- **ID-51 – Datenbank-Leistung & Dimensionierung** – Auswirkung eigener Abfragen auf die Leistung
- **ID-60 – API-Referenz** – die Schnittstelle als Alternative zum direkten Zugriff

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL VI · BETRIEB

Produktsicherheit & Meldewege

Schwachstellen melden, Aktualisierungen, sichere Ausserbetriebnahme

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-53 Produktsicherheit

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Produktsicherheit & Meldewege

Dieses Kapitel enthält die Angaben nach **Anhang II der Cyberresilienz-Verordnung** (Verordnung (EU) 2024/2847). Es richtet sich an **IT-Betrieb** und an alle, die über die Beschaffung entscheiden. Die technische Installation steht in ID-50, die Härtung des Servers ebenfalls dort.

Dieselben Angaben stehen öffentlich unter ordivis.eu/produktsicherheit.html.

INHALT

1 Kontaktstelle für Schwachstellen
2 Unterstützungszeitraum
3 Sicherer Erstbetrieb
4 Sicherheitsaktualisierungen einspielen
5 Selbsttätiges Einspielen abschalten
6 Auswirkungen von Änderungen auf Ihre Daten
7 Sichere Außerbetriebnahme und Datenlöschung
8 Stückliste der Fremdbestandteile
9 Bekannte Umstände, die zu erheblichen Risiken führen können
10 EU-Konformitätserklärung
11 Hersteller
Verwandte Themen

1 Kontaktstelle für Schwachstellen

Melden Sie Schwachstellen an **security@ordivis.eu**. Das ist der einzige von uns benannte Meldeweg; maschinenlesbar steht er unter `https://ordivis.eu/.well-known/security.txt`.

Bitte veröffentlichen Sie Ihre Erkenntnisse nicht, solange eine Abhilfe aussteht. Unsere Strategie zur koordinierten Offenlegung, die Reaktionszeiten und die Zusage an gutgläubige Finder stehen auf der oben genannten Seite.

Ist eine Schwachstelle aktiv ausgenutzt, sind wir seit dem 11. September 2026 nach Artikel 14 der Verordnung verpflichtet, sie binnen 24 Stunden dem zuständigen Computer-Notfallteam und der ENISA zu melden. Wir unterrichten Sie dann ebenfalls — über die im Vertrag hinterlegte technische Anschrift. **Halten Sie diese Anschrift aktuell:** Sie ist der Weg, auf dem Sie von uns erfahren, dass Handlungsbedarf besteht.

2 Unterstützungszeitraum

Bis einschließlich 31. Dezember 2032. Bis zu diesem Zeitpunkt behandeln wir Schwachstellen und stellen Sicherheitsaktualisierungen **kostenlos** bereit. Der Zeitraum gilt für alle bis dahin ausgelieferten Fassungen und besteht unabhängig vom vereinbarten Supportmodell.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

3 Sicherer Erstbetrieb

Ordivis ist für den Betrieb in einem verwalteten Firmennetz hinter einer Firewall vorgesehen, auf einem gepflegten Windows-Server mit PostgreSQL. Vor der Freigabe für Benutzer:

1. **Verschlüsselte Verbindung sicherstellen.** Im Auslieferungszustand läuft der Server in der Betriebsart `ReverseProxy` und erwartet, dass die Verschlüsselung **vorgelagert** abgeschlossen wird. Wird er ohne diese betrieben, laufen Anmeldedaten unverschlüsselt über das Netz. Alternativ die Betriebsart `Certificate` (eigenes Zertifikat) oder `Acme` (automatischer Bezug) einstellen — siehe ID-50.
2. **Erstkonto absichern.** Das beim Onboarding angelegte Konto erhält ein starkes Kennwort; Mehrfaktoranmeldung aktivieren (ID-41).
3. **Ungenutzte Module abschalten.** Jedes deaktivierte Modul ist eine Angriffsfläche weniger (ID-40).
4. **Erkennung mit eigenen Konten betreiben.** Für Netzscans eigene Konten mit den **geringsten ausreichenden Rechten** vergeben, keine Domänenadministratoren (Abschnitt 9).
5. **Datensicherung einrichten und einen Rücklauf prüfen** (ID-35). Eine Sicherung, die nie zurückgespielt wurde, ist eine Vermutung.

4 Sicherheitsaktualisierungen einspielen

Aktualisierungen werden über `https://ordivis.eu/updates` bereitgestellt. Jedes Paket trägt ein **signiertes Manifest**; der Aktualisierungsdienst prüft die Signatur vor dem Einspielen und bricht bei Abweichung ab.

- **Server:** Der Aktualisierungsvorgang ist in ID-50 beschrieben.
- **Arbeitsplätze:** Der Windows-Dienst `InfraDesk.ClientUpdate` vergleicht die installierte Fassung mit der bereitgestellten und spielt Aktualisierungen ein.

Prüfen Sie nach jeder Aktualisierung die Fassung unter **Über** in der Anwendung.

5 Selbsttätiges Einspielen abschalten

Sie können Aktualisierungen von Hand steuern. Halten Sie dazu den Dienst an und setzen Sie den Starttyp auf *Deaktiviert*:

```
Stop-Service InfraDesk.ClientUpdate
Set-Service  InfraDesk.ClientUpdate -StartupType Disabled
```

Zum Wiedereinschalten `-StartupType Automatic` setzen und den Dienst starten.

> **Was Sie damit übernehmen.** Ohne diesen Dienst erreichen Sie **auch Sicherheitsaktualisierungen nicht mehr selbsttätig**. Sie müssen dann selbst verfolgen, welche Fassungen erscheinen, und sie einspielen. Unsere Pflicht, sie bereitzustellen, bleibt bestehen — die Pflicht, sie anzuwenden, liegt dann vollständig bei Ihnen.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

6 Auswirkungen von Änderungen auf Ihre Daten

- **Datenbankwanderungen** laufen beim Aktualisieren des Servers. Sie sind **nicht rückwärts** ausführbar: Ein Rücksprung auf eine ältere Fassung erfordert das Rückspielen einer Sicherung. **Vor jeder Aktualisierung sichern.**
- **Mandanten** liegen in getrennten Datenbanken. Das Löschen eines Mandanten entfernt dessen Datenbank vollständig.
- **Modulschalter** blenden Funktionen aus, **löschen aber keine Daten**. Ein später wieder eingeschaltetes Modul findet seinen Bestand vor.
- **Rechteänderungen** wirken sofort auf Lesen und Schreiben; bereits erzeugte Exporte und Berichte sind davon nicht berührt.

7 Sichere Außerbetriebnahme und Datenlöschung

Beim Rückbau einer Anlage sind **fünf** Orte zu räumen. Die Deinstallation über die Systemsteuerung entfernt die Dienste und Programmdateien — **die Daten bleiben absichtlich erhalten**, damit ein Rückbau nicht versehentlich zum Datenverlust wird.

1. **Dienste anhalten und entfernen:** `InfraDesk.API`, `InfraDesk.Portal`, `InfraDesk.ClientUpdate`, an entfernten Standorten `InfraDesk.Discovery.Collector`.
2. **Geplante Aufgaben entfernen:** `Ordivis-DailyBackup` und `Ordivis-Restore` (in Bestandsanlagen `InfraDesk-DailyBackup`, `InfraDesk-Restore`).
3. **Datenbanken löschen.** Je Mandant eine Datenbank, dazu die Verwaltungsdatenbank. Die PostgreSQL-Instanz heißt in Standardinstallationen `postgresql-ordivis`.
4. **Konfiguration und Sicherungen entfernen:** `%ProgramData%\InfraDesk` — darin liegen Konfiguration, hinterlegte Geheimnisse und, sofern nicht anders eingestellt, die Sicherungen.
5. **Protokolle entfernen.** ⚠️ **Die Protokolle liegen nicht zwingend unter `%ProgramData%`.** Ist in `%ProgramData%\InfraDesk\config\operations.json` ein abweichender Pfad hinterlegt, gilt dieser — er kann auf ein anderes Laufwerk zeigen. **Lesen Sie diese Datei, bevor Sie sie löschen**, sonst bleiben Protokolldaten zurück.

Auf den Arbeitsplätzen zusätzlich den Client deinstallieren und dessen lokales Benutzerprofilverzeichnis entfernen.

> **Sicheres Löschen.** Das Entfernen von Dateien und Datenbanken gibt den Speicher frei, überschreibt ihn aber nicht. Wo das gefordert ist — etwa bei Datenträgern, die das Haus verlassen — setzen Sie die Mittel Ihres Betriebssystems oder Ihrer Speicherplattform ein.

8 Stückliste der Fremdbestandteile

Jedem Installationspaket liegen im Programmverzeichnis bei:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Datei	Inhalt
`ordivis-sbom.cdx.json`	Maschinenlesbare Stückliste im Format **CycloneDX 1.6**
`THIRD-PARTY-NOTICES.txt`	Urhebervermerke und vollständige Lizenztexte

Die Stückliste nennt jeden ausgelieferten Fremdbestandteil mit Fassung, Lizenz und Verwendungszweck. Sie beantwortet die Frage, ob eine bekannt gewordene Schwachstelle in einer Fremdkomponente Ihre Anlage betrifft.

9 Bekannte Umstände, die zu erheblichen Risiken führen können

Anhang II Nummer 5 verlangt, solche Umstände zu benennen. Wir tun das vollständig.

- **Betrieb ohne vorgelagerte Verschlüsselung.** Siehe Abschnitt 3 Punkt 1. Der häufigste und folgenschwerste Fehler bei der Inbetriebnahme.
- **Die Erkennung speichert Anmeldedaten für Netzgeräte**, die häufig privilegiert sind. Wer Zugriff auf den Ordivis-Server erlangt, erlangt mittelbar Zugriff auf diese Daten.
- **Ordivis führt eine vollständige Abbildung Ihrer IT-Landschaft.** Ein unbefugter Lesezugriff ist deshalb für einen Angreifer ein Aufklärungsgewinn — auch ohne jede Schreibmöglichkeit.
- **Die Absicherung von Betriebssystem, Datenbank und Sicherungen liegt bei Ihnen.** Ordivis kann eine unsichere Umgebung nicht ausgleichen.

10 EU-Konformitätserklärung

Liegt noch nicht vor. Die Konformitätsbewertung nach der Cyberresilienz-Verordnung ist zum **11. Dezember 2027** fällig; die Erklärung wird zu diesem Zeitpunkt unter ordivis.eu/produktsicherheit.html veröffentlicht. Eine CE-Kennzeichnung für Cybersicherheit führt Ordivis Platform bis dahin nicht.

11 Hersteller

Grams IT, Christian Grams, Mittelstraße 18, 34277 Fuldabrück, Deutschland. Allgemeine Anschrift **info@ordivis.eu**, Schwachstellen **security@ordivis.eu**.

Verwandte Themen

- **ID-50 – Betriebshandbuch – Installation & Serverbetrieb** – Sicherer Erstbetrieb und Härtung
- **ID-40 – Administrationshandbuch** – Aktualisierungen einspielen und Jobs steuern
- **ID-35 – Modul Datensicherung** – Sicherungen vor Änderungen und vor der Außerbetriebnahme
- **ID-02 – Systemvoraussetzungen & Architektur** – Komponenten und Netzwege des Produkts

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TECHNISCHE REFERENZ

API-Referenz

REST-API – Authentifizierung, Konventionen, Endpunkte und Nutzung

Produktversion 2026.08.19 · Handbuch-Revision 1.0

Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-60 API-Referenz

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

API-Referenz

Technisches Referenzdokument für Fachpersonal (Entwickler, Integratoren, Automatisierung). Es beschreibt die REST-API von Ordivis Platform: Authentifizierung, Autorisierung, Konventionen, Fehlerbehandlung und die verfügbaren Endpunkt-Bereiche. Das Rechtemodell ist in **ID-44** vertieft.

INHALT

1 Überblick
2 Authentifizierung
- 2.1 Anmeldung
- 2.2 Token verwenden
- 2.3 Erneuern & Mandant wechseln
- 2.4 Maschinen-Zugang (Integrations-Token)

3 Autorisierung
4 Konventionen
5 Fehlerbehandlung
6 Endpunkt-Bereiche
7 Beispiele
- 7.1 Anmelden (curl)
- 7.2 Liste abrufen (mit Token)
- 7.3 Ressource erstellen

8 Nutzungshinweise & Betrieb
Verwandte Themen

1 Überblick

- **Stil:** REST über HTTPS, JSON als Austauschformat.
- **Basis-URL:** `https://<host>/` (Standard-Port 443). Alle fachlichen Endpunkte liegen unter `/api/v1/...`.
- **Versionierung:** URL-basiert (`/api/v1`). Neue, inkompatible Versionen erhalten einen neuen Pfad (`/api/v2`). Innerhalb einer Major-Version bleiben Endpunkte abwärtskompatibel.
- **Interaktive Dokumentation:** Die API stellt **OpenAPI/Swagger** bereit (Swagger-UI unter `/swagger`). Dort finden Sie die exakten Request-/Response-Schemata jedes Endpunkts.

> **Hinweis:** Dieses Dokument beschreibt Struktur und Nutzung. Die **maschinenlesbare, stets aktuelle** Schnittstellenbeschreibung ist die OpenAPI-Spezifikation der jeweils installierten Version.

2 Authentifizierung

Ordivis Platform verwendet **JWT (Bearer-Token)**.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

2.1 Anmeldung

`POST /api/v1/auth/central-login`

Request-Body:

```
{
  "login": "christian.grams",
  "password": "GeheimesKennwort",
  "tenantId": null
}
```

Response (`LoginResult`):

```
{
  "accessToken": "eyJhbGciOiJI … ",
  "refreshToken": "b3f1 … ",
  "expiresAt": "2026-07-11T12:30:00+00:00",
  "tenantId": "1e2d … ",
  "tenantName": "Musterstadt",
  "requiresTenantSelection": false,
  "tenants": []
}
```

- Der **Benutzername** ist case-insensitiv, das **Kennwort** case-sensitiv.
- Gehört der Benutzer zu **mehreren Mandanten** und wurde keiner gewählt, ist `requiresTenantSelection = true` und `tenants` enthält die Auswahl. Wiederholen Sie den Login mit gesetztem `tenantId` .

2.2 Token verwenden

Senden Sie den Access-Token bei jedem Aufruf im Header:

```
Authorization: Bearer <accessToken>
```

Der Token enthält die **`permission`-Claims** des Benutzers (siehe ID-44) sowie den Mandantenkontext.

2.3 Erneuern & Mandant wechseln

- **Erneuern:** Mit dem `refreshToken` einen neuen Access-Token anfordern (Refresh-Endpunkt der Auth-Ressource), bevor `expiresAt` erreicht ist.
- **Mandant wechseln:** `POST /api/v1/auth/switch-tenant` stellt für Benutzer mit mehreren Mandanten einen Token für den Zielmandanten aus (erfordert entsprechendes Recht, siehe ID-42).

2.4 Maschinen-Zugang (Integrations-Token)

Externe Werkzeuge und der Discovery-Collector melden sich **nicht** mit einem Benutzer-Login an, sondern mit einem **Integrations-Token** – einem widerrufbaren Maschinen-Zugang, der zu genau einem Mandanten gehört. Er reist im selben Header:

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

```
Authorization: Bearer ifd_<zufälliger Wert>
```

Unterschiede zum Benutzer-Token, die im Betrieb zählen:

- Ein Integrations-Token **läuft nicht ab** (sofern kein Ablaufdatum gesetzt wurde) und braucht kein Refresh – ein Dienst am Standort soll nach einem Stromausfall von selbst wiederkommen.
- Er trägt **nur die Einspeise-Rechte**, die bei seiner Ausstellung festgelegt wurden. Neue Rechte wachsen einem bestehenden Token **nicht** zu; für eine erweiterte Fläche wird ein neuer ausgestellt.
- Die Ingest-Endpunkte akzeptieren **ausschließlich** dieses Verfahren. Ein angemeldeter Benutzer erreicht sie auch mit dem passenden Recht nicht – und umgekehrt kommt ein Integrations-Token nicht an die Fachendpunkte der Oberfläche. Die beiden Flächen sind getrennt, nicht abgestuft.

Ausstellung und Widerruf beschreibt das Betriebshandbuch (**ID-50**, Abschnitt 7.4).

3 Autorisierung

- Jeder geschützte Endpunkt verlangt eine bestimmte **Berechtigung** (`[RequiresPermission("…")]`).
- Fehlt die Berechtigung → **HTTP 403 Forbidden**.
- Ohne (gültigen) Token → **HTTP 401 Unauthorized**.
- Die vollständige Rechte-Referenz steht in **ID-44**.

4 Konventionen

Thema	Regel
Content-Type	`application/json; charset=utf-8` für Request und Response
IDs	GUID (UUID) als String
Datum/Zeit	ISO 8601; Zeitstempel in **UTC** (`+00:00`). Eingehende Werte werden nach UTC normalisiert
Paginierung	Listen über `page` / `pageSize` (bzw. `skip` / `take`) als Query-Parameter; Antwort enthält Gesamtanzahl
Filter/Sortierung	Query-Parameter je Endpunkt (Filterfelder, `sort` / `order`)
Mandantenkontext	ergibt sich aus dem Token; keine Mandanten-ID im Body nötig
Idempotenz	`PUT` / `DELETE` idempotent; `POST` erzeugt neue Ressourcen

Wichtig: Zeitwerte müssen als UTC bzw. mit Offset `+00:00` übergeben werden. Ein Offset ungleich Null wird von der Datenbank (timestamptz) abgelehnt – die API normalisiert clientseitige Werte vor dem Speichern nach UTC.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

5 Fehlerbehandlung

HTTP-Status	Bedeutung
200 / 201 / 204	Erfolg (OK / Erstellt / Kein Inhalt)
400	Ungültige Anfrage (z. B. Validierungsfehler, falsches Datumsformat)
401	Nicht authentifiziert (Token fehlt/abgelaufen)
403	Authentifiziert, aber Berechtigung fehlt
402	**Lizenz-/Aktivierungsgrenze** – schreibende Aufrufe bei abgelaufener Lizenz oder Admin-Limit (siehe ID-43)
404	Ressource nicht gefunden
409	Konflikt (z. B. eindeutige Verletzung, veralteter Stand)
422	Nicht verarbeitbar (semantische Validierung)
429	Zu viele Anfragen (Rate-Limit, z. B. E-Mail-Versand)
500	Serverfehler

Fehlerantworten liefern einen strukturierten Body (Problem-Details) mit Kurzbeschreibung.

6 Endpunkt-Bereiche

Die API ist modular geschnitten. Übersicht der Basis-Routen (Details in Swagger):

Bereich	Basis-Route	Zweck
Authentifizierung	`/api/v1/auth`	Login, Token-Refresh, Mandantenwechsel
Ersteinrichtung	`/api/v1/setup`	First-Run-Onboarding
Tickets / Service Desk	`/api/v1/tickets`	Tickets, dazu Unterrouten (s. u.)
Changes	`/api/v1/changes`	Change Requests
Probleme	`/api/v1/problems`	Problem-Management
CMDB	`/api/v1/cmdb`	CIs, `…/inventory`, `…/processes`, `…/label-media` (Etikettenmedien), `…/labels/print` (Druckauftrag je Druckersprache), `…/nfc-tags` (NFC-Zuordnung)
IPAM	`/api/v1/ipam`	Subnetze, IP-Adressen, DHCP
Discovery	`/api/v1/discovery`, `/api/v1/scan-credentials`	Scans, Anmeldedaten, Collector, Client-Inventar

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Bereich	Basis-Route	Zweck
Maschinen-Naht (Ingest)	`/api/v1/ipam/ingest`, `/api/v1/discovery/ingest`, `/api/v1/discovery/collector`, `/api/v1/security/ingest`	Aufnahme externer Agenten und Collector (nur Integrations-Token, s. u.)
Assets/Verträge	`/api/v1/contracts`, `/api/v1/vendors`	Verträge, Lieferanten
Wissensdatenbank	`/api/v1/tickets/knowledge`, `/api/v1/portal/knowledge`	KB intern / Portal
Sicherheit (ISMS)	`/api/v1/security`, `/api/v1/security/oscal`, `/api/v1/security/profiles`	BSI/ISO, OSCAL-Export
BCM	`/api/v1/bcm`	Business Continuity
DR-Planung	`/api/v1/drplans`	Notfall-/Wiederanlaufpläne
Workflow	`/api/v1/workflow`	Workflows & Aufgaben
Reporting	`/api/reporting`, `/api/v1/dashboards`	Berichte, Dashboards
Checklisten	`/api/v1/checklists`	Vorlagen und Durchführungen
Handbücher	`/api/v1/handbooks`	Betriebs-/Notfall-/Wiederanlaufhandbücher
Organisation & GVP	`/api/v1/organisation`	Aufbau-/Ablauforganisation, `…/catalog`, `…/gvp`, `…/ozg`, `…/fim`
Datenschutz (VVT)	`/api/v1/datenschutz`	Verarbeitungstätigkeiten, `…/datenpannen`, `…/avv`, `…/kataloge`, `…/export`, `…/dashboard`
Betroffenenrechte	`/api/v1/privacy`	Auskunft, Löschung, Berichtigung (Art. 15–22 DSGVO)
Audit	`/api/v1/audit`	Audit-Log; `…/export` gibt die gefilterten Einträge als CSV aus (`audit.export`)
Benachrichtigungen	`/api/v1/notifications`	In-App-Benachrichtigungen
E-Mail	`/api/v1/mail/accounts`	Postfächer (Versand/Eingang)
Identität	`/api/v1/auth`, `/api/v1/identity/mfa`, `/api/identity/directory`, `/api/identity/access-reviews`	Benutzer, Zwei-Faktor, Verzeichnis, Access Reviews
SSO (SAML)	`/sso/saml`	Anmeldung über einen externen Identitätsanbieter

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Bereich	Basis-Route	Zweck
Mandanten/Plattform	`/api/v1/platform/tenants`, `/api/v1/platform/grants`	Mandanten, Cross-Tenant-Grants
Lizenz/Aktivierung	`/api/v1/licenses`, `/api/v1/activation`	Produktlizenzierung
System	`/api/v1/system`, `/api/v1/settings`, `/api/v1/jobs`, `/api/system/backup`, `/api/system/logging`, `/api/v1/system/ha`	Betrieb & Konfiguration
Darstellung	`/api/v1/branding`	Firmenname und Logo für erzeugte Dokumente (je Mandant)
Wallboard	`/api/v1/wallboard`	Anzeigetafel-Zugang (eigenes Token-Verfahren, keine Anmeldung)
Mobile App	`/api/v1/mobile`	Naht für die Android-App (Inventur, Etiketten, Warteschlange)
Support	`/api/v1/support`	Fehlerbericht aus dem Windows-Client
Suche	`/api/search`	Übergreifende Suche
Anhänge	`/api/v1/attachments`	Datei-Anhänge

Ticket-Unterrouten (Auszug): `sla-profiles`, `routing-rules`, `approval-tiers`, `maintenance-windows`, `canned-responses`, `templates`, `support-groups`, `contacts`, `reporting`, `{ticketId}/knowledge`, `{ticketId}/render`. Daneben stehen `/api/v1/approval-delegations` (Vertretung bei Freigaben) und `/api/v1/ticketing/signatures` (Signaturen unter ausgehenden Mails).

7 Beispiele

7.1 Anmelden (curl)

```
curl -X POST https://host/api/v1/auth/central-login \
  -H "Content-Type: application/json" \
  -d '{"login":"christian.grams","password":"***"}'
```

7.2 Liste abrufen (mit Token)

```
curl https://host/api/v1/tickets?page=1&pageSize=50 \
  -H "Authorization: Bearer <accessToken>"
```

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

7.3 Ressource erstellen

```
curl -X POST https://host/api/v1/tickets \
  -H "Authorization: Bearer <accessToken>" \
  -H "Content-Type: application/json" \
  -d '{ "title":"Drucker im 2. OG", "urgency":"High", "impact":"Medium", "categoryId":" ... "
```

Achtung: `urgency` und `impact` akzeptieren nur `Low` / `Medium` / `High`. „Kritisch“ ist eine **abgeleitete Priorität**, kein Eingabewert (führt sonst zu HTTP 400).

8 Nutzungshinweise & Betrieb

- **TLS ist Pflicht.** Betreiben Sie Integrationen nie über unverschlüsselte Verbindungen.
- **Least Privilege:** Legen Sie für Automatisierung ein **eigenes Dienstkonto** mit genau den benötigten Rechten an (ID-44), nicht einen Administrator.
- **Rate-Limits:** Insbesondere versendende Aktionen sind ratenbegrenzt (HTTP 429). Implementieren Sie Backoff/Retry.
- **Stabilität:** Verlassen Sie sich auf `/api/v1`; prüfen Sie vor Upgrades die OpenAPI-Diffs.
- **Zeitzonen:** Immer UTC senden und empfangen; lokale Darstellung erst im Client.

Verwandte Themen

- **ID-44 – Rollen- & Rechte-Matrix** – Rechte des technischen Kontos
- **ID-42 – Modul Mandantenfähigkeit** – Mandantenauswahl im Aufruf
- **ID-52 – Datenbank-Referenz** – Datenstruktur hinter den Endpunkten
- **ID-43 – Modul Produktlizenzierung** – lizenzabhängige Limits

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Ordivis

TEIL VII · ANHANG

Anhang – Glossar & Referenzen

Glossar, Rechte-Referenz, Tastenkürzel und Versionshistorie

Produktversion 2026.08.19 · Handbuch-Revision 1.0
Stand 19.08.2026 · Plattform .NET 10 / Windows Server

ID-90 Anhang

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten. Vervielfältigung nur mit schriftlicher Genehmigung.

Anhang – Glossar & Referenzen

Rund um IT-Service-Management sammeln sich viele Abkürzungen und Fachbegriffe an, und nicht jeder kennt sie aus dem Stand. Dieser Anhang ist zum Nachschlagen gedacht: Er erklärt die im Handbuch verwendeten Begriffe kurz und knapp, listet die wichtigsten Berechtigungen und Tastenkürzel und hält die Versionshistorie fest. Sie müssen ihn nicht am Stück lesen – schlagen Sie nach, wenn Ihnen ein Begriff begegnet, den Sie einordnen möchten.

INHALT

A Glossar
B Rechte-Referenz (Auswahl)
C Tastenkürzel & Bedienhinweise
D Konventionen
E Support
F Versionshistorie des Handbuchs
G Stichwortverzeichnis
H Software-Stückliste (SBOM)
Software-Stückliste (SBOM)
Verwandte Themen

A Glossar

Begriff	Bedeutung
Asset	Vermögenswert (Gerät, Vertrag, Lizenz) mit Kosten-/Lebenszyklus-Sicht.
BAO	Besondere Aufbauorganisation – Notfallorganisation im BCM.
BCM	Business Continuity Management – Geschäftsfortführung.
BIA	Business-Impact-Analyse – Kritikalität und Zeitziele je Prozess.
CAB	Change Advisory Board – Gremium zur Change-Freigabe.
CI	Configuration Item – dokumentierter IT-Baustein in der CMDB.
CMDB	Configuration Management Database.
CSAT	Customer Satisfaction – Zufriedenheitsbewertung.
DR	Disaster Recovery – technischer Wiederanlauf.
Incident	Störung; Ziel: schnelle Wiederherstellung.
IPAM	IP Address Management – IP-/Netzverwaltung.
ISMS	Informationssicherheits-Managementsystem.
KEDB	Known Error Database – bekannte Fehler mit Workaround.
Known Error	Bekannter Fehler mit dokumentiertem Workaround.
Mandant	Abgegrenzte Organisationseinheit; Daten sind getrennt.
MFA	Mehr-Faktor-Authentifizierung.
MTPD	Maximal tolerierbare Ausfallzeit.
OLA	Operational Level Agreement – interne Zulieferzeiten.
Problem	Ursache hinter wiederkehrenden Incidents.
RBAC	Role-Based Access Control – rollenbasierte Rechtevergabe.
RPO	Recovery Point Objective – tolerierbarer Datenverlust.
RTO	Recovery Time Objective – Wiederanlaufzeit.
SACM	Service Asset & Configuration Management (ITIL).
Service Request	Standardauftrag, ggf. mit Genehmigung.
SLA	Service Level Agreement – vereinbarte Service-Zeiten.

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

B Rechte-Referenz (Auswahl)

> **Vollständig steht der Katalog in ID-44.** Diese Auswahl nennt je Bereich ein Beispiel zur Orientierung; maßgeblich sind die 140 Rechte der Rollen- und Rechte-Matrix. Wer eine Rolle zusammenstellt, arbeitet mit ID-44 – nicht mit dieser Tabelle.

Recht (Beispiel)	Bereich
`system.config.manage`	Administrator / Systemkonfiguration
`system.backup.manage`	Sicherung und Wiederherstellung
`identity.users.manage`	Benutzer & Rollen
`tenant.manage`	Mandanten
`tickets.read` / `tickets.read.all` / `tickets.update`	Service Desk
`cmdb.ci.read` / `cmdb.ci.update`	CMDB
`ipam.prefix.read` / `ipam.ip.assign`	IPAM
`cmdb.ci.lifecycle.manage` / `contracts.manage`	Assets (als CI geführt) & Verträge
`discovery.manage`	Discovery-Scans
`knowledge.read.internal` / `knowledge.author` / `knowledge.publish`	Wissensdatenbank
`workflow.read` / `workflow.manage`	Workflows
`reporting.read` / `reporting.manage`	Reporting
`audit.read` / `audit.export`	Audit-Log
`licensing.manage`	Produktlizenz

C Tastenkürzel & Bedienhinweise

- **Doppelklick** in Listen öffnet den Detaildatensatz.
- **Spalten** lassen sich ziehen (umordnen), in der Breite ändern und über die Spaltenauswahl ein-/ausblenden.
- **Export**: In jeder Tabelle über das Export-Symbol als CSV oder PDF.
- **Verknüpfte Objekte** (z. B. CI im Ticket) sind anklickbar und führen zur Detailansicht.

D Konventionen

- Menüpfade mit ▸, Feld-/Schaltflächennamen *kursiv*, technische Werte in `Festbreitenschrift`.
- Es werden stets **Klarnamen** angezeigt, nie interne IDs/GUIDs.
- **Acht Symbole**, in jedem Dokument mit derselben Bedeutung:

Symbol	Bedeutung
🖥️	der Weg im Windows-Client
🌐	der Weg im Web-Portal
⚙️	der Weg am Server – Konsole, Konfigurationsdatei, Datenbank
🔑	das Recht, das die Handlung verlangt
▸	trennt die Stationen eines Menüpfads
🔒	Pflichtreihenfolge – die Schritte bauen aufeinander auf
💡	empfohlene Reihenfolge – Abweichen kostet Nacharbeit
✅	Ergebnis – woran Sie erkennen, dass es geklappt hat

- Jede Handlungsanweisung folgt demselben Aufbau: **Anleitungskopf** (Oberfläche + Recht) → Voraussetzung → nummerierte Schritte → Ergebnis. Die Langfassung mit Beispielen steht in ID-00, Abschnitt *Konventionen und Symbole*.
- Steht vor einer Schrittfolge **keine** Reihenfolge-Marke, ist die Reihenfolge frei. Zählt sie, steht 🔒 oder 💡 da – Schweigen bedeutet nie „vermutlich egal".

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

E Support

Support erfolgt ausschließlich per **E-Mail und Ticket**. Reaktionszeiten: **kritische Störungen innerhalb von 8 Stunden**, sonstige Anfragen **innerhalb von 3 Arbeitstagen**.

F Versionshistorie des Handbuchs

Revision	Datum	Änderung
1.0	2026-07-10	Erstausgabe: modulare Dokumentation, Corporate Design (MD3), alle Module.
1.1	2026-08-09	Dokumentenkarte vollständig (36 Dokumente), Abschnitt *Verwandte Themen* in allen Dokumenten, Stichwortverzeichnis, Inhaltsverzeichnis und Kapitel-Lesezeichen im PDF.

G Stichwortverzeichnis

Das Verzeichnis nennt zu jedem Stichwort die **Hauptfundstelle** – das Dokument, das den Begriff im Zusammenhang behandelt – und weitere Dokumente, in denen er eine Rolle spielt. Beiläufige Erwähnungen bleiben außen vor. Die Fundstellen werden beim Bau aus den Quellen ermittelt; sie können nicht auf ein Dokument zeigen, in dem der Begriff nicht vorkommt.

Stichwort	Hauptfundstelle	Auch in
Abhängigkeit	**ID-30**	ID-01, ID-28a
Active Directory	**ID-21b**	ID-24, ID-41, ID-40, ID-02
Anmeldung	**ID-10**	ID-41, ID-60, ID-01, ID-50
Asset	**ID-23**	ID-21, ID-52, ID-01, ID-43
Attribut	**ID-21**	ID-41, ID-21b, ID-26, ID-40
Audit-Log	**ID-27**	ID-40, ID-01, ID-44
Aufgabe	**ID-34**	ID-26, ID-40, ID-01, ID-00
Auswirkungsanalyse	**ID-29**	ID-21
Backup	**ID-35**	ID-50, ID-53, ID-52, ID-30
BAO (Besondere Aufbauorganisation)	**ID-29**	ID-44
Barrierefreiheit	**ID-12**	ID-10
BCM	**ID-29**	ID-28, ID-44, ID-52, ID-28a
Benachrichtigung	**ID-31**	ID-10, ID-35, ID-40, ID-20
Benutzerkonto	**ID-21b**	ID-41, ID-21, ID-40, ID-50
Berechtigung	**ID-44**	ID-27a, ID-60, ID-02, ID-41
Beziehung (CI)	**ID-21**	ID-30, ID-33
BIA (Business-Impact-Analyse)	**ID-29**	ID-30, ID-44
BPMN	**ID-21**	ID-21a
Branding	**ID-40**	ID-27a
CAB (Change Advisory Board)	**ID-20**	ID-26, ID-11
Change	**ID-20**	ID-10, ID-44, ID-29a, ID-26
Checkliste	**ID-29a**	ID-44, ID-52, ID-29, ID-30
CI (Configuration Item)	**ID-21**	ID-21b, ID-34, ID-30, ID-40
CI-Klasse	**ID-21**	ID-40, ID-21c, ID-21b
CMDB	**ID-21**	ID-21b, ID-23, ID-44, ID-52
CSV-Import	**ID-21a**	–
Dashboard	**ID-27**	ID-22, ID-01, ID-60, ID-27a
Datenqualität	**ID-21**	ID-01

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Stichwort	Hauptfundstelle	Auch in
Datenschutz	**ID-32**	ID-10, ID-40, ID-44, ID-52
DHCP	**ID-22**	ID-44, ID-52, ID-50
Discovery	**ID-24**	ID-21, ID-44, ID-52, ID-02
DR (Disaster Recovery)	**ID-30**	ID-29, ID-29a, ID-28, ID-00
DSGVO	**ID-32**	ID-44, ID-28a, ID-10
Eskalation	**ID-20**	ID-11, ID-29, ID-27, ID-34
Etikett	**ID-21**	ID-01, ID-21c, ID-60, ID-40
Export	**ID-32**	ID-40, ID-28, ID-27, ID-27a
Farbschema	**ID-12**	–
Fehlerbericht	**ID-40**	–
Freigabe	**ID-25**	ID-40, ID-32, ID-22, ID-44
Genehmigung	**ID-26**	ID-10, ID-20, ID-01, ID-31
Geschäftsprozess	**ID-21a**	ID-21, ID-29, ID-28a, ID-33
Geschäftsverteilungsplan	**ID-33**	ID-44, ID-21a, ID-60
Glocke (Benachrichtigungen)	**ID-31**	–
Härtung	**ID-50**	ID-53, ID-02, ID-00
Hochkontrast	**ID-12**	–
IMAP	**ID-31**	ID-40
Impressum	**ID-00**	ID-31
Incident	**ID-11**	ID-28a, ID-01, ID-20, ID-10
Installation	**ID-50**	ID-42, ID-02, ID-43, ID-53
Inventar	**ID-24**	ID-21, ID-23, ID-01
IPAM	**ID-22**	ID-44, ID-52, ID-01, ID-60
ISMS	**ID-28**	ID-28a, ID-52, ID-01, ID-29
ISO/IEC 27001	**ID-28**	ID-28a, ID-00, ID-44
ITIL	**ID-20**	ID-01
Job (geplante Aufgabe)	**ID-40**	ID-52, ID-53
Kategorie	**ID-20**	ID-32, ID-11, ID-29a, ID-25
KEDB	**ID-25**	ID-20, ID-01
Kennwort	**ID-41**	ID-10, ID-01, ID-24, ID-50
LDAP	**ID-24**	ID-41, ID-40, ID-21b, ID-02
Lebenszyklus	**ID-23**	ID-20, ID-28a, ID-21, ID-01
Lizenz (Produkt)	**ID-43**	ID-02, ID-60, ID-29a, ID-23
Mandant	**ID-42**	ID-40, ID-51, ID-44, ID-60
MFA	**ID-41**	ID-60, ID-01, ID-44, ID-52
Migration	**ID-50**	ID-21c, ID-43, ID-40, ID-52
Notfallplan	**ID-29**	–
OLA	**ID-20**	–
PostgreSQL	**ID-51**	ID-50, ID-02, ID-53, ID-40
Priorität	**ID-11**	ID-20, ID-27a, ID-10, ID-20a
Problem-Management	**ID-20**	–
Prozessdokumentation	**ID-21**	ID-44

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Stichwort	Hauptfundstelle	Auch in
RACI	**ID-21a**	ID-21, ID-33
RBAC	**ID-41**	ID-44, ID-27, ID-02
Report	**ID-27a**	ID-27, ID-52, ID-44, ID-00
Risiko	**ID-28**	ID-28a, ID-32, ID-29, ID-44
Rolle	**ID-41**	ID-44, ID-27, ID-29, ID-42
RPO	**ID-30**	ID-29, ID-35, ID-52
RTO	**ID-30**	ID-29, ID-35, ID-52
Scan (Discovery)	**ID-22**	ID-24, ID-21, ID-50, ID-02
Schutzbedarf	**ID-28**	ID-32, ID-28a
Selbstorganisation	**ID-34**	ID-26, ID-01, ID-00
Self-Service-Portal	**ID-10**	ID-12, ID-32, ID-34, ID-50
Service-Katalog	**ID-10**	ID-44
SLA	**ID-20**	ID-11, ID-27, ID-44, ID-34
SMTP	**ID-31**	ID-50
SNMP	**ID-24**	ID-02
SoA (Erklärung zur Anwendbarkeit)	**ID-28**	ID-28a, ID-01, ID-52
SQL-Abfrage	**ID-27a**	ID-00, ID-44, ID-52
SSH	**ID-24**	ID-02
SSO	**ID-40**	ID-60
Standortbaum	**ID-21**	ID-24, ID-01, ID-28a
Statuswechsel	**ID-29a**	ID-20
Subnetz	**ID-22**	ID-44, ID-50
Suche	**ID-25**	ID-60, ID-21, ID-20
Support	**ID-40**	ID-11, ID-20, ID-50, ID-02
Tastenkürzel	**ID-12**	–
Ticket	**ID-20**	ID-11, ID-34, ID-25, ID-10
TLS/Zertifikat	**ID-31**	ID-50, ID-02, ID-28a, ID-34
Topologie	**ID-21**	–
Update	**ID-50**	ID-40, ID-44, ID-11, ID-00
Verarbeitungstätigkeit	**ID-32**	ID-44, ID-60, ID-40, ID-52
Vertrag	**ID-23**	ID-52, ID-44, ID-27a, ID-26
Vorgangsnummer	**ID-20**	–
Wallboard	**ID-27**	ID-50, ID-60
Wartungsfenster	**ID-20a**	ID-34, ID-27, ID-20, ID-44
Wiedervorlage	**ID-20**	ID-32, ID-11, ID-34
Wissensartikel	**ID-25**	ID-10, ID-44, ID-01, ID-11
WMI	**ID-24**	ID-02
Workflow	**ID-26**	ID-44, ID-52, ID-60, ID-29a
Zeitplan	**ID-35**	ID-27a, ID-40, ID-50
Zugriffsrecht	**ID-27a**	ID-44, ID-50

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

H Software-Stückliste (SBOM)

Software-Stückliste (SBOM)

Alle Bestandteile dieser Auslieferung mit Version, Herkunft und Lizenz. Dieselben Angaben zeigt der Windows-Client und das Web-Portal unter *Systemstatus › Software-Stückliste (SBOM)*; sie liegen der Installation zusätzlich als `THIRD-PARTY-NOTICES.txt` und `ordivis-sbom.cdx.json` (CycloneDX 1.6) bei.

Fassung dieser Auslieferung: `2026.8.19.1452`

PLATTFORM UND LAUFZEIT

Maßgeblich ist hier nicht das Alter, sondern das Support-Ende: Eine Anlage, deren Datenbank oder Laufzeit aus der Pflege gefallen ist, ist kein Technikthema mehr, sondern ein Betriebsrisiko.

Name	Version	Support-Ende	Lizenz	Verwendung
.NET	`10.0`	2028-11	MIT	Laufzeitgrundlage aller Serverdienste und des Windows-Clients
ASP.NET Core	`10.0.10`	2028-11	MIT	Webrahmenwerk des Serverdienstes (Schnittstelle, Web-Portal)
Windows Desktop Runtime	`10`	2028-11	MIT	Laufzeit des Fortschrittsfensters beim Update (WPF)
PostgreSQL	`18.4-1`	2030-11	PostgreSQL	Datenbank je Mandant; wird beim Serversetup mitinstalliert
Windows App SDK	`1.6.250602001`	*nicht ausgewiesen*	MIT	Oberflaechengrundlage des Windows-Clients und des Wallboards (WinUI 3)
Rebus	`8.9.2`	*nicht ausgewiesen*	MIT	Nachrichtenbus fuer Hintergrundvorgaenge
Inno Setup	`6`	*nicht ausgewiesen*	Inno Setup License	Erzeugt die Installationspakete (nur Bauwerkzeug, wird nicht ausgeliefert)
WiX Toolset	`7.0.0`	*nicht ausgewiesen*	MS-RL	Erzeugt das MSI-Paket fuer die Softwareverteilung (nur Bauwerkzeug)

FREMDBIBLIOTHEKEN — DIREKT EINGEBUNDEN (88)

Name	Version	Lizenz	Verwendung
Asp.Versioning.Mvc	`10.0.0`	MIT	Versionierung der Programmierschnittstelle (/api/v1)
AspNetCore.HealthChecks.NpgSql	`9.0.0`	Apache-2.0	Bereitschaftspruefung der Datenbankverbindung
BouncyCastle.Cryptography	`2.7.0`	MIT	Kryptographische Verfahren fuer Zertifikate und Signaturen
Certes	`3.0.4`	MIT	Automatischer Bezug von TLS-Zertifikaten ueber ACME (Let's Encrypt)
ClosedXML	`0.105.1`	MIT	Erzeugt und liest Excel-Mappen (Berichte, Rasterexport, Importvorlagen)
CommunityToolkit.Mvvm	`8.4.2`	MIT	Grundgeruest der Ansichtsmodelle im Windows-Client
CommunityToolkit.WinUI.UI.Controls.DataGrid	`7.1.2`	MIT	Tabellenraster im Windows-Client
CsvHelper	`33.1.0`	Apache-2.0	Liest und schreibt CSV-Dateien (Import und Export)
FluentValidation	`11.12.0`	Apache-2.0	Pruefung von Eingaben an der Schnittstelle
FluentValidation.DependencyInjectionExtensions	`11.12.0`	Apache-2.0	Anbindung der Eingabepruefung
HtmlSanitizer	`9.0.892`	MIT	Entfernt gefaehrliche Bestandteile aus fremdem HTML (E-Mail-Rumpf, Wissensartikel)
Isopoh.Cryptography.Argon2	`2.0.0`	siehe beigelegten Lizenztext	Hashen der Kennwoerter lokaler Benutzerkonten (Argon2)
Lextm.SharpSnmpLib	`12.5.7`	MIT	SNMP-Abfrage von Netzgeraeten bei der Erkennung
LiveChartsCore.SkiaSharpView.WinUI	`2.0.0-rc4.5`	MIT	Diagramme in den Uebersichten des Windows-Clients
MailKit	`4.17.0`	MIT	Versand und Abruf von E-Mail (SMTP, IMAP, POP3)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Markdig	`0.37.0`	BSD-2-Clause	Wandelt Markdown in HTML (Wissensdatenbank, Handbuchbausteine)
MediatR	`12.5.0`	Apache-2.0	Innere Vermittlung von Befehlen und Abfragen
Microsoft.AspNetCore.Authentication.Cookies	`2.3.12`	Apache-2.0	Anmeldung im Web-Portal ueber Sitzungscookie
Microsoft.AspNetCore.Authentication.JwtBearer	`10.0.10`	MIT	Anmeldung an der Schnittstelle ueber Token
Microsoft.AspNetCore.Authentication.Negotiate	`10.0.10`	MIT	Windows-Anmeldung (Einmalanmeldung)
Microsoft.AspNetCore.Authentication.OpenIdConnect	`10.0.10`	MIT	Anmeldung ueber einen fremden Ausweisdienst
Microsoft.AspNetCore.SignalR.Client	`10.0.10`	MIT	Sofortmeldungen an Client und Portal
Microsoft.EntityFrameworkCore	`10.0.10`	MIT	Datenzugriff und Datenbankwanderungen
Microsoft.EntityFrameworkCore.Design	`10.0.10`	MIT	Werkzeugunterstuetzung fuer Datenbankwanderungen
Microsoft.EntityFrameworkCore.Relational	`10.0.10`	MIT	Datenzugriff, relationaler Teil
Microsoft.Extensions.Caching.Abstractions	`10.0.10`	MIT	Schnittstellen der Zwischenspeicherung
Microsoft.Extensions.Caching.Memory	`10.0.10`	MIT	Zwischenspeicher im Arbeitsspeicher
Microsoft.Extensions.Configuration	`10.0.10`	MIT	Einlesen der Konfiguration
Microsoft.Extensions.Configuration.Abstractions	`10.0.10`	MIT	Schnittstellen der Konfiguration
Microsoft.Extensions.Configuration.Binder	`10.0.10`	MIT	Zuordnung der Konfiguration auf Objekte
Microsoft.Extensions.Configuration.EnvironmentVariables	`10.0.10`	MIT	Konfiguration aus Umgebungsvariablen
Microsoft.Extensions.Configuration.Json	`10.0.10`	MIT	Konfiguration aus JSON-Dateien (appsettings)
Microsoft.Extensions.DependencyInjection	`10.0.10`	MIT	Verwaltung der Programmbausteine
Microsoft.Extensions.DependencyInjection.Abstractions	`8.0.2`	MIT	Schnittstellen dazu
Microsoft.Extensions.Hosting	`10.0.10`	MIT	Lebenszyklus des Serverdienstes
Microsoft.Extensions.Hosting.Abstractions	`10.0.10`	MIT	Schnittstellen dazu
Microsoft.Extensions.Hosting.WindowsServices	`10.0.10`	MIT	Betrieb als Windows-Dienst
Microsoft.Extensions.Http	`10.0.10`	MIT	Verwaltung ausgehender HTTP-Verbindungen
Microsoft.Extensions.Logging.Abstractions	`8.0.3`	MIT	Schnittstellen der Protokollierung
Microsoft.Extensions.Logging.Debug	`10.0.10`	MIT	Protokollausgabe beim Entwickeln
Microsoft.Extensions.Options	`10.0.10`	MIT	Typisierter Zugriff auf Einstellungen
Microsoft.Extensions.Options.ConfigurationExtensions	`10.0.10`	MIT	Anbindung der Einstellungen an die Konfiguration
Microsoft.Graphics.Win2D	`1.3.0`	MIT	Zeichenflaeche des Windows-Clients
Microsoft.Maui.Controls	`10.0.20`	MIT	Oberflaechengeruest der Android-App
Microsoft.Win32.Registry	`5.0.0`	MIT	Lesen der Windows-Registrierung bei der Erkennung
Microsoft.Windows.SDK.BuildTools	`10.0.26100.1742`	ungeklaert	Nur zur Bauzeit — wird nicht ausgeliefert
Microsoft.WindowsAppSDK	`1.6.250602001`	siehe beigelegten Lizenztext	Laufzeitgrundlage des Windows-Clients (WinUI)
MsgReader	`6.1.0`	MIT	Liest Outlook-.msg-Dateien als Ticket-Anhang
MudBlazor	`7.16.0`	MIT	Oberflaechenbausteine des Web-Portals
Npgsql	`10.0.3`	PostgreSQL	Datenbanktreiber fuer PostgreSQL
Npgsql.EntityFrameworkCore.PostgreSQL	`10.0.3`	PostgreSQL	PostgreSQL-Anbindung des Datenzugriffs
QRCoder	`1.8.0`	MIT	Erzeugt QR-Etiketten fuer Gegenstaende
QuestPDF	`2024.12.3`	siehe beigelegten Lizenztext	Erzeugt die PDF-Dokumente (Berichte, Notfallhandbuch, Nachweise)

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Rebus	`8.9.2`	MIT	Nachrichtenbus fuer Hintergrundvorgaenge
Rebus.ServiceProvider	`10.7.2`	MIT	Anbindung des Nachrichtenbusses
ReverseMarkdown	`5.4.0`	MIT	Wandelt HTML zurueck in Markdown (E-Mail-Aufnahme)
Serilog	`4.4.0`	Apache-2.0	Betriebsprotokollierung
Serilog.AspNetCore	`10.0.0`	Apache-2.0	Betriebsprotokollierung im Serverdienst
Serilog.Extensions.Hosting	`10.0.0`	Apache-2.0	Anbindung der Protokollierung an den Dienst
Serilog.Extensions.Logging	`10.0.0`	Apache-2.0	Anbindung an die .NET-Protokollierung
Serilog.Formatting.Compact	`3.0.0`	Apache-2.0	Protokollausgabe im kompakten JSON-Format
Serilog.Settings.Configuration	`10.0.1`	Apache-2.0	Protokolleinstellungen aus der Konfiguration
Serilog.Sinks.EventLog	`4.0.0`	Apache-2.0	Schreibt Protokolleintraege ins Windows-Ereignisprotokoll
Serilog.Sinks.File	`7.0.0`	Apache-2.0	Schreibt Protokolleintraege in Dateien
sqlite-net-pcl	`1.9.172`	siehe beigelegten Lizenztext	Lokale Zwischenspeicherung in der Android-App
SQLitePCLRaw.bundle_green	`2.1.11`	Apache-2.0	SQLite-Laufzeit der Android-App
SQLitePCLRaw.lib.e_sqlite3.android	`2.1.12`	Apache-2.0	SQLite-Laufzeit der Android-App
SSH.NET	`2026.0.0`	MIT	SSH-Zugriff auf Netzgeraete bei der Erkennung
Swashbuckle.AspNetCore	`6.9.0`	MIT	Erzeugt die OpenAPI-Beschreibung der Schnittstelle
System.Diagnostics.EventLog	`10.0.10`	MIT	Schreiben in das Windows-Ereignisprotokoll
System.DirectoryServices.Protocols	`10.0.10`	MIT	LDAP-Abgleich mit dem Verzeichnisdienst
System.IO.Packaging	`10.0.0`	MIT	Lesen und Schreiben von Office-Paketformaten
System.Management	`10.0.10`	MIT	Abfrage von Windows-Systemdaten bei der Erkennung
System.Security.Cryptography.ProtectedData	`10.0.10`	MIT	Schutz lokal abgelegter Geheimnisse
System.Security.Cryptography.Xml	`10.0.10`	MIT	Signaturpruefung in XML-Dokumenten
Xamarin.AndroidX.Activity.Ktx	`1.13.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Collection.Ktx	`1.6.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Fragment.Ktx	`1.8.9.3`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Lifecycle.LiveData	`2.11.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Lifecycle.LiveData.Core.Ktx	`2.11.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Lifecycle.Process	`2.11.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Lifecycle.Runtime.Ktx	`2.11.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Lifecycle.Runtime.Ktx.Android	`2.11.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.Lifecycle.ViewModel.Ktx	`2.11.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
Xamarin.AndroidX.SavedState.SavedState.Ktx	`1.5.0.1`	MIT AND Apache-2.0	*nicht ausgewiesen*

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.Firebase.Messaging	`125.1.1`	MIT AND Apache-2.0	*nicht ausgewiesen*
ZXing.Net	`0.16.11`	Apache-2.0	Erzeugt und liest Strichcodes
ZXing.Net.Maui.Controls	`0.10.3`	MIT	Strichcode-Erfassung mit der Kamera in der Android-App

FREMDBIBLIOTHEKEN — MITTELBAR EINGEBUNDEN (250)

Mittelbar eingebundene Bestandteile werden vollständig ausgewiesen, nicht zusammengefasst: Schwachstellen sitzen erfahrungsgemäß gerade dort.

Name	Version	Lizenz	Verwendung
AngleSharp	`0.17.1`	MIT	wird von AngleSharp.Css, HtmlSanitizer benoetigt
AngleSharp.Css	`0.17.0`	MIT	wird von HtmlSanitizer benoetigt
Asp.Versioning.Abstractions	`10.0.0`	MIT	wird von Asp.Versioning.Http benoetigt
Asp.Versioning.Http	`10.0.0`	MIT	wird von Asp.Versioning.Mvc benoetigt
ClosedXML.Parser	`2.0.0`	MIT	wird von ClosedXML benoetigt
DocumentFormat.OpenXml	`3.1.1`	MIT	wird von ClosedXML benoetigt
DocumentFormat.OpenXml.Framework	`3.1.1`	MIT	wird von DocumentFormat.OpenXml benoetigt
ExcelNumberFormat	`1.1.0`	MIT	wird von ClosedXML benoetigt
GoogleGson	`2.13.1.1`	MIT AND Apache-2.0	wird von Xamarin.Google.Crypto.Tink.Android benoetigt
HarfBuzzSharp	`7.3.0.3-preview.2.2`	MIT	wird von SkiaSharp.HarfBuzz benoetigt
HarfBuzzSharp.NativeAssets.macOS	`7.3.0.3-preview.2.2`	MIT	wird von HarfBuzzSharp benoetigt
HarfBuzzSharp.NativeAssets.Win32	`7.3.0.3-preview.2.2`	MIT	wird von HarfBuzzSharp benoetigt
HtmlAgilityPack	`1.12.4`	MIT	wird von ReverseMarkdown benoetigt
Humanizer.Core	`2.14.1`	MIT	wird von Microsoft.CodeAnalysis.CSharp.Workspaces, Microsoft.CodeAnalysis.Workspaces.Common, Microsoft.CodeAnalysis.Workspaces.MSBuild benoetigt
Isopoh.Cryptography.Blake2b	`2.0.0`	siehe beigelegten Lizenztext	wird von Isopoh.Cryptography.Argon2 benoetigt
Isopoh.Cryptography.SecureArray	`2.0.0`	siehe beigelegten Lizenztext	wird von Isopoh.Cryptography.Argon2, Isopoh.Cryptography.Blake2b benoetigt
LiveChartsCore	`2.0.0-rc4.5`	MIT	wird von LiveChartsCore.Behaviours, LiveChartsCore.SkiaSharpView, LiveChartsCore.SkiaSharpView.WinUI benoetigt
LiveChartsCore.Behaviours	`2.0.0-rc4.5`	MIT	wird von LiveChartsCore.SkiaSharpView.WinUI benoetigt
LiveChartsCore.SkiaSharpView	`2.0.0-rc4.5`	MIT	wird von LiveChartsCore.SkiaSharpView.WinUI benoetigt
MediatR.Contracts	`2.0.1`	Apache-2.0	wird von MediatR benoetigt
Microsoft.AspNetCore.App.Internal.Assets	`10.0.11`	MIT	indirekt eingebunden
Microsoft.AspNetCore.Connections.Abstractions	`10.0.10`	MIT	wird von Microsoft.AspNetCore.Http.Connections.Common, Microsoft.AspNetCore.SignalR.Common benoetigt
Microsoft.AspNetCore.Http.Connections.Client	`10.0.10`	MIT	wird von Microsoft.AspNetCore.SignalR.Client benoetigt
Microsoft.AspNetCore.Http.Connections.Common	`10.0.10`	MIT	wird von Microsoft.AspNetCore.Http.Connections.Client benoetigt
Microsoft.AspNetCore.SignalR.Client.Core	`10.0.10`	MIT	wird von Microsoft.AspNetCore.SignalR.Client benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Microsoft.AspNetCore.SignalR.Common	`10.0.10`	MIT	wird von Microsoft.AspNetCore.SignalR.Client.Core, Microsoft.AspNetCore.SignalR.Protocols.Json benoetigt
Microsoft.AspNetCore.SignalR.Protocols.Json	`10.0.10`	MIT	wird von Microsoft.AspNetCore.SignalR.Client.Core benoetigt
Microsoft.Bcl.Cryptography	`10.0.2`	MIT	wird von Microsoft.IdentityModel.Tokens benoetigt
Microsoft.Build.Framework	`18.0.2`	MIT	wird von Microsoft.CodeAnalysis.Workspaces.MSBuild, Microsoft.EntityFrameworkCore.Design benoetigt
Microsoft.CodeAnalysis.Analyzers	`3.11.0`	MIT	wird von Microsoft.CodeAnalysis.CSharp, Microsoft.CodeAnalysis.CSharp.Workspaces, Microsoft.CodeAnalysis.Common benoetigt
Microsoft.CodeAnalysis.Common	`5.0.0`	MIT	wird von Microsoft.CodeAnalysis.CSharp, Microsoft.CodeAnalysis.CSharp.Workspaces, Microsoft.CodeAnalysis.Workspaces.Common benoetigt
Microsoft.CodeAnalysis.CSharp	`5.0.0`	MIT	wird von Microsoft.CodeAnalysis.CSharp.Workspaces, Microsoft.EntityFrameworkCore.Design benoetigt
Microsoft.CodeAnalysis.CSharp.Workspaces	`5.0.0`	MIT	wird von Microsoft.EntityFrameworkCore.Design benoetigt
Microsoft.CodeAnalysis.Workspaces.Common	`5.0.0`	MIT	wird von Microsoft.CodeAnalysis.CSharp.Workspaces, Microsoft.CodeAnalysis.Workspaces.MSBuild benoetigt
Microsoft.CodeAnalysis.Workspaces.MSBuild	`5.0.0`	MIT	wird von Microsoft.EntityFrameworkCore.Design benoetigt
Microsoft.EntityFrameworkCore.Abstractions	`10.0.10`	MIT	wird von Microsoft.EntityFrameworkCore benoetigt
Microsoft.EntityFrameworkCore.Analyzers	`10.0.10`	MIT	wird von Microsoft.EntityFrameworkCore benoetigt
Microsoft.Extensions.ApiDescription.Server	`6.0.5`	MIT	wird von Swashbuckle.AspNetCore benoetigt
Microsoft.Extensions.Configuration.CommandLine	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting benoetigt
Microsoft.Extensions.Configuration.FileExtensions	`10.0.10`	MIT	wird von Microsoft.Extensions.Configuration.Json, Microsoft.Extensions.Hosting benoetigt
Microsoft.Extensions.Configuration.UserSecrets	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting benoetigt
Microsoft.Extensions.DependencyModel	`10.0.10`	MIT	wird von Microsoft.EntityFrameworkCore.Design, Serilog.Settings.Configuration benoetigt
Microsoft.Extensions.Diagnostics	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting, Microsoft.Extensions.Http benoetigt
Microsoft.Extensions.Diagnostics.Abstractions	`10.0.10`	MIT	wird von Microsoft.Extensions.Diagnostics, Microsoft.Extensions.Hosting.Abstractions benoetigt
Microsoft.Extensions.Features	`10.0.10`	MIT	wird von Microsoft.AspNetCore.Connections.Abstractions benoetigt
Microsoft.Extensions.FileProviders.Abstractions	`10.0.10`	MIT	wird von Microsoft.Extensions.Configuration.FileExtensions, Microsoft.Extensions.Configuration.Json, Microsoft.Extensions.Configuration.UserSecrets benoetigt
Microsoft.Extensions.FileProviders.Physical	`10.0.10`	MIT	wird von Microsoft.Extensions.Configuration.FileExtensions, Microsoft.Extensions.Configuration.UserSecrets, Microsoft.Extensions.Hosting benoetigt
Microsoft.Extensions.FileSystemGlobbing	`10.0.10`	MIT	wird von Microsoft.Extensions.FileProviders.Physical benoetigt
Microsoft.Extensions.Logging	`10.0.10`	MIT	wird von Microsoft.AspNetCore.SignalR.Client.Core, Microsoft.CodeAnalysis.Workspaces.MSBuild, Microsoft.EntityFrameworkCore benoetigt
Microsoft.Extensions.Logging.Configuration	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting, Microsoft.Extensions.Logging.Console benoetigt
Microsoft.Extensions.Logging.Console	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting benoetigt
Microsoft.Extensions.Logging.EventLog	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting, Microsoft.Extensions.Hosting.WindowsServices benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Microsoft.Extensions.Logging.EventSource	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting benoetigt
Microsoft.Extensions.Primitives	`10.0.10`	MIT	wird von Microsoft.CodeAnalysis.Workspaces.MSBuild, Microsoft.Extensions.Caching.Abstractions, Microsoft.Extensions.Caching.Memory benoetigt
Microsoft.IdentityModel.Abstractions	`8.19.2`	MIT	wird von Microsoft.IdentityModel.Logging benoetigt
Microsoft.IdentityModel.JsonWebTokens	`8.19.2`	MIT	wird von System.IdentityModel.Tokens.Jwt benoetigt
Microsoft.IdentityModel.Logging	`8.19.2`	MIT	wird von Microsoft.IdentityModel.Tokens benoetigt
Microsoft.IdentityModel.Protocols	`8.19.2`	MIT	wird von Microsoft.IdentityModel.Protocols.OpenIdConnect benoetigt
Microsoft.IdentityModel.Protocols.OpenIdConnect	`8.19.2`	MIT	wird von Microsoft.AspNetCore.Authentication.JwtBearer, Microsoft.AspNetCore.Authentication.OpenIdConnect benoetigt
Microsoft.IdentityModel.Tokens	`8.19.2`	MIT	wird von Microsoft.IdentityModel.JsonWebTokens, Microsoft.IdentityModel.Protocols, System.IdentityModel.Tokens.Jwt benoetigt
Microsoft.IO.RecyclableMemoryStream	`3.0.1`	MIT	wird von MsgReader benoetigt
Microsoft.Maui.Controls.Build.Tasks	`10.0.20`	MIT	wird von Microsoft.Maui.Controls benoetigt
Microsoft.Maui.Controls.Core	`10.0.20`	MIT	wird von Microsoft.Maui.Controls, Microsoft.Maui.Controls.Build.Tasks, Microsoft.Maui.Controls.Xaml benoetigt
Microsoft.Maui.Controls.Xaml	`10.0.20`	MIT	wird von Microsoft.Maui.Controls, Microsoft.Maui.Controls.Build.Tasks benoetigt
Microsoft.Maui.Core	`10.0.20`	MIT	wird von Microsoft.Maui.Controls.Core, Microsoft.Maui.Controls.Xaml benoetigt
Microsoft.Maui.Essentials	`10.0.20`	MIT	wird von Microsoft.Maui.Core benoetigt
Microsoft.Maui.Graphics	`10.0.80`	MIT	wird von Microsoft.Maui.Core, Microsoft.Maui.Essentials, MsgReader benoetigt
Microsoft.Maui.Resizetizer	`10.0.20`	MIT	wird von Microsoft.Maui.Controls benoetigt
Microsoft.NET.ILLink.Tasks	`10.0.11`	MIT	indirekt eingebunden
Microsoft.OpenApi	`1.6.14`	MIT	wird von Swashbuckle.AspNetCore.Swagger benoetigt
Microsoft.VisualStudio.SolutionPersistence	`1.0.52`	MIT	wird von Microsoft.CodeAnalysis.Workspaces.MSBuild benoetigt
Microsoft.Web.WebView2	`1.0.2651.64`	siehe beigelegten Lizenztext	wird von Microsoft.WindowsAppSDK benoetigt
Microsoft.Win32.SystemEvents	`6.0.0`	MIT	wird von System.Drawing.Common benoetigt
MimeKit	`4.17.0`	MIT	wird von MailKit benoetigt
Mono.TextTemplating	`3.0.0`	MIT	wird von Microsoft.EntityFrameworkCore.Design benoetigt
Newtonsoft.Json	`13.0.4`	MIT	wird von Certes, Microsoft.CodeAnalysis.Workspaces.MSBuild, Microsoft.EntityFrameworkCore.Design benoetigt
OpenMcdf	`3.1.4`	MPL-2.0	wird von MsgReader benoetigt
Portable.BouncyCastle	`1.9.0`	MIT	wird von Certes benoetigt
RBush.Signed	`4.0.0`	MIT	wird von ClosedXML benoetigt
RtfPipe	`2.0.7677.4303`	MIT	wird von MsgReader benoetigt
Serilog.Sinks.Console	`6.1.1`	Apache-2.0	wird von Serilog.AspNetCore benoetigt
Serilog.Sinks.Debug	`3.0.0`	Apache-2.0	wird von Serilog.AspNetCore benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
SixLabors.Fonts	`1.0.0`	Apache-2.0	wird von ClosedXML benoetigt
SkiaSharp	`2.88.9-preview.2.2`	MIT	wird von LiveChartsCore.SkiaSharpView, SkiaSharp.HarfBuzz, SkiaSharp.Views.WinUI benoetigt
SkiaSharp.HarfBuzz	`2.88.9-preview.2.2`	MIT	wird von LiveChartsCore.SkiaSharpView, LiveChartsCore.SkiaSharpView.WinUI benoetigt
SkiaSharp.NativeAssets.macOS	`2.88.9-preview.2.2`	MIT	wird von SkiaSharp benoetigt
SkiaSharp.NativeAssets.Win32	`2.88.9-preview.2.2`	MIT	wird von SkiaSharp benoetigt
SkiaSharp.NativeAssets.WinUI	`2.88.9-preview.2.2`	MIT	wird von SkiaSharp.Views.WinUI benoetigt
SkiaSharp.Views.WinUI	`2.88.9-preview.2.2`	MIT	wird von LiveChartsCore.SkiaSharpView.WinUI benoetigt
SQLitePCLRaw.core	`2.1.11`	Apache-2.0	wird von SQLitePCLRaw.provider.e_sqlite3 benoetigt
SQLitePCLRaw.provider.e_sqlite3	`2.1.11`	Apache-2.0	wird von SQLitePCLRaw.bundle_green benoetigt
Square.OkIO	`3.17.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Core.OkIO, Xamarin.AndroidX.DataStore.Preferences.Core benoetigt
Square.OkIO.JVM	`3.17.0.1`	MIT AND Apache-2.0	wird von Square.OkIO, Xamarin.AndroidX.DataStore.Android, Xamarin.AndroidX.DataStore.Core.OkIO.Jvm benoetigt
Swashbuckle.AspNetCore.Swagger	`6.9.0`	MIT	wird von Swashbuckle.AspNetCore, Swashbuckle.AspNetCore.SwaggerGen benoetigt
Swashbuckle.AspNetCore.SwaggerGen	`6.9.0`	MIT	wird von Swashbuckle.AspNetCore benoetigt
Swashbuckle.AspNetCore.SwaggerUI	`6.9.0`	MIT	wird von Swashbuckle.AspNetCore benoetigt
System.CodeDom	`6.0.0`	MIT	wird von Mono.TextTemplating, System.Management benoetigt
System.Composition	`9.0.0`	MIT	wird von Microsoft.CodeAnalysis.CSharp.Workspaces, Microsoft.CodeAnalysis.Workspaces.Common, Microsoft.CodeAnalysis.Workspaces.MSBuild benoetigt
System.Composition.AttributedModel	`9.0.0`	MIT	wird von System.Composition, System.Composition.Convention, System.Composition.TypedParts benoetigt
System.Composition.Convention	`9.0.0`	MIT	wird von System.Composition benoetigt
System.Composition.Hosting	`9.0.0`	MIT	wird von System.Composition, System.Composition.TypedParts benoetigt
System.Composition.Runtime	`9.0.0`	MIT	wird von System.Composition, System.Composition.Hosting, System.Composition.TypedParts benoetigt
System.Composition.TypedParts	`9.0.0`	MIT	wird von System.Composition benoetigt
System.Drawing.Common	`6.0.0`	MIT	wird von QRCoder benoetigt
System.IdentityModel.Tokens.Jwt	`8.19.2`	MIT	wird von Microsoft.IdentityModel.Protocols.OpenIdConnect benoetigt
System.Security.Cryptography.Pkcs	`10.0.9`	MIT	wird von MimeKit, MsgReader, System.Security.Cryptography.Xml benoetigt
System.ServiceProcess.ServiceController	`10.0.10`	MIT	wird von Microsoft.Extensions.Hosting.WindowsServices benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
UTF.Unknown	`2.6.0`	MPL-1.1 [Wahl aus: MPL-1.1 OR GPL-2.0-or-later OR LGPL-2.1-or-later]	wird von MsgReader benoetigt
Xamarin.Android.Glide	`4.16.0.14`	MIT AND BSD-2-Clause AND Apache-2.0	wird von Microsoft.Maui.Core benoetigt
Xamarin.Android.Glide.Annotations	`4.16.0.14`	MIT AND BSD-2-Clause AND Apache-2.0	wird von Xamarin.Android.Glide benoetigt
Xamarin.Android.Glide.DiskLruCache	`4.16.0.14`	MIT AND BSD-2-Clause AND Apache-2.0	wird von Xamarin.Android.Glide benoetigt
Xamarin.Android.Glide.GifDecoder	`4.16.0.14`	MIT AND BSD-2-Clause AND Apache-2.0	wird von Xamarin.Android.Glide benoetigt
Xamarin.AndroidX.Activity	`1.13.0.1`	MIT AND Apache-2.0	wird von Microsoft.Maui.Essentials, Xamarin.AndroidX.Activity.Ktx, Xamarin.AndroidX.AppCompat benoetigt
Xamarin.AndroidX.Annotation	`1.10.0.1`	MIT AND Apache-2.0	wird von Xamarin.Android.Glide.GifDecoder, Xamarin.AndroidX.Activity, Xamarin.AndroidX.AppCompat benoetigt
Xamarin.AndroidX.Annotation.Experimental	`1.6.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Browser, Xamarin.AndroidX.Camera.Core, Xamarin.AndroidX.Camera.View benoetigt
Xamarin.AndroidX.Annotation.Jvm	`1.10.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Annotation, Xamarin.AndroidX.Collection.Jvm, Xamarin.AndroidX.DataStore.Android benoetigt
Xamarin.AndroidX.AppCompat	`1.7.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.View, Xamarin.AndroidX.ConstraintLayout, Xamarin.AndroidX.Navigation.UI benoetigt
Xamarin.AndroidX.AppCompat.AppCompatResources	`1.7.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat benoetigt
Xamarin.AndroidX.Arch.Core.Common	`2.2.0.21`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Arch.Core.Runtime, Xamarin.AndroidX.Lifecycle.LiveData, Xamarin.AndroidX.Lifecycle.LiveData.Core benoetigt
Xamarin.AndroidX.Arch.Core.Runtime	`2.2.0.21`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.LiveData, Xamarin.AndroidX.Lifecycle.LiveData.Core, Xamarin.AndroidX.Lifecycle.Runtime.Android benoetigt
Xamarin.AndroidX.Browser	`1.8.0.11`	MIT AND Apache-2.0	wird von Microsoft.Maui.Essentials benoetigt
Xamarin.AndroidX.Camera.Camera2	`1.4.2.3`	MIT AND Apache-2.0	wird von ZXing.Net.Maui benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.AndroidX.Camera.Core	`1.4.2.3`	MIT AND Apache-2.0 AND BSD-3-Clause	wird von Xamarin.AndroidX.Camera.Camera2, Xamarin.AndroidX.Camera.Lifecycle, Xamarin.AndroidX.Camera.Video benoetigt
Xamarin.AndroidX.Camera.Lifecycle	`1.4.2.3`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.View, ZXing.Net.Maui benoetigt
Xamarin.AndroidX.Camera.Video	`1.4.2.3`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.View benoetigt
Xamarin.AndroidX.Camera.View	`1.4.2.3`	MIT AND Apache-2.0	wird von ZXing.Net.Maui benoetigt
Xamarin.AndroidX.CardView	`1.0.0.36`	MIT AND Apache-2.0	wird von Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.Collection	`1.6.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.AppCompat.AppCompatResources, Xamarin.AndroidX.Browser benoetigt
Xamarin.AndroidX.Collection.Jvm	`1.6.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Collection, Xamarin.AndroidX.Collection.Ktx, Xamarin.AndroidX.Lifecycle.Runtime.Android benoetigt
Xamarin.AndroidX.Compose.Runtime.Annotation	`1.11.3.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.NavigationEvent benoetigt
Xamarin.AndroidX.Compose.Runtime.Annotation.Android	`1.11.3.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.NavigationEvent.Android benoetigt
Xamarin.AndroidX.Concurrent.Futures	`1.3.0.4`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Browser, Xamarin.AndroidX.Camera.Camera2, Xamarin.AndroidX.Camera.Core benoetigt
Xamarin.AndroidX.Concurrent.Futures.Ktx	`1.3.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.Core, Xamarin.AndroidX.Camera.Lifecycle benoetigt
Xamarin.AndroidX.ConstraintLayout	`2.2.1.3`	MIT AND Apache-2.0	wird von Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.ConstraintLayout.Core	`1.1.1.3`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.ConstraintLayout benoetigt
Xamarin.AndroidX.CoordinatorLayout	`1.3.0.3`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Navigation.UI, Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.Core	`1.19.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.AppCompat.AppCompatResources, Xamarin.AndroidX.Browser benoetigt
Xamarin.AndroidX.Core.Core.Ktx	`1.19.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.CustomView.PoolingContainer benoetigt
Xamarin.AndroidX.Core.ViewTree	`1.0.0.6`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.Core, Xamarin.AndroidX.Lifecycle.Runtime.Android benoetigt
Xamarin.AndroidX.CursorAdapter	`1.0.0.34`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat benoetigt
Xamarin.AndroidX.CustomView	`1.2.0.4`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.CoordinatorLayout, Xamarin.AndroidX.DrawerLayout, Xamarin.AndroidX.Navigation.UI benoetigt
Xamarin.AndroidX.CustomView.PoolingContainer	`1.1.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.RecyclerView benoetigt
Xamarin.AndroidX.DataStore	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences benoetigt
Xamarin.AndroidX.DataStore.Android	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore, Xamarin.AndroidX.DataStore.Preferences.Android benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.AndroidX.DataStore.Core	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore, Xamarin.AndroidX.DataStore.Core.OkIO, Xamarin.AndroidX.DataStore.Preferences.Core benoetigt
Xamarin.AndroidX.DataStore.Core.Android	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Android, Xamarin.AndroidX.DataStore.Core, Xamarin.AndroidX.DataStore.Core.Jvm benoetigt
Xamarin.AndroidX.DataStore.Core.Jvm	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Core.OkIO.Jvm benoetigt
Xamarin.AndroidX.DataStore.Core.OkIO	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore, Xamarin.AndroidX.DataStore.Preferences.Core benoetigt
Xamarin.AndroidX.DataStore.Core.OkIO.Jvm	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Android, Xamarin.AndroidX.DataStore.Core.OkIO, Xamarin.AndroidX.DataStore.Preferences.Core.Android benoetigt
Xamarin.AndroidX.DataStore.Preferences	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Common, Xamarin.Firebase.Installations benoetigt
Xamarin.AndroidX.DataStore.Preferences.Android	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences benoetigt
Xamarin.AndroidX.DataStore.Preferences.Core	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences benoetigt
Xamarin.AndroidX.DataStore.Preferences.Core.Android	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences.Android benoetigt
Xamarin.AndroidX.DataStore.Preferences.External.Protobuf	`1.2.1.1`	MIT AND BSD-3-Clause	wird von Xamarin.AndroidX.DataStore.Preferences.Proto benoetigt
Xamarin.AndroidX.DataStore.Preferences.Proto	`1.2.1.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences.Core.Android benoetigt
Xamarin.AndroidX.DocumentFile	`1.1.0.4`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Legacy.Support.Core.Utils benoetigt
Xamarin.AndroidX.DrawerLayout	`1.2.0.18`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.Navigation.UI, Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.DynamicAnimation	`1.1.0.3`	MIT AND Apache-2.0	wird von Microsoft.Maui.Controls.Core, Xamarin.AndroidX.Transition, Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.Emoji2	`1.5.0.6`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.Emoji2.ViewsHelper benoetigt
Xamarin.AndroidX.Emoji2.ViewsHelper	`1.5.0.6`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat benoetigt
Xamarin.AndroidX.ExifInterface	`1.4.2`	MIT AND Apache-2.0	wird von Xamarin.Android.Glide, Xamarin.AndroidX.Camera.Core, ZXing.Net.Maui benoetigt
Xamarin.AndroidX.Fragment	`1.8.9.3`	MIT AND Apache-2.0	wird von Xamarin.Android.Glide, Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.Fragment.Ktx benoetigt
Xamarin.AndroidX.Interpolator	`1.0.0.37`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Browser, Xamarin.AndroidX.Core, Xamarin.AndroidX.SwipeRefreshLayout benoetigt
Xamarin.AndroidX.Legacy.Support.Core.Utils	`1.0.0.37`	MIT AND Apache-2.0	wird von Xamarin.GooglePlayServices.Stats benoetigt
Xamarin.AndroidX.Lifecycle.Common	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.Camera.Core, Xamarin.AndroidX.Camera.Lifecycle benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.AndroidX.Lifecycle.Common.Jvm	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.Common, Xamarin.AndroidX.Lifecycle.Runtime.Android, Xamarin.AndroidX.Lifecycle.ViewModelSavedState.Android benoetigt
Xamarin.AndroidX.Lifecycle.LiveData.Core	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Fragment, Xamarin.AndroidX.Lifecycle.LiveData, Xamarin.AndroidX.Lifecycle.LiveData.Core.Ktx benoetigt
Xamarin.AndroidX.Lifecycle.Runtime	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.Core benoetigt
Xamarin.AndroidX.Lifecycle.Runtime.Android	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.Runtime, Xamarin.AndroidX.Lifecycle.Runtime.Ktx.Android, Xamarin.AndroidX.Navigation.Common.Android benoetigt
Xamarin.AndroidX.Lifecycle.ViewModel	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.Fragment benoetigt
Xamarin.AndroidX.Lifecycle.ViewModel.Android	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.ViewModel, Xamarin.AndroidX.Lifecycle.ViewModelSavedState.Android, Xamarin.AndroidX.Navigation.Common.Android benoetigt
Xamarin.AndroidX.Lifecycle.ViewModelSavedState	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.Fragment, Xamarin.AndroidX.Navigation.Common benoetigt
Xamarin.AndroidX.Lifecycle.ViewModelSavedState.Android	`2.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.ViewModelSavedState, Xamarin.AndroidX.Navigation.Common.Android benoetigt
Xamarin.AndroidX.Loader	`1.1.0.37`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Fragment, Xamarin.AndroidX.Legacy.Support.Core.Utils benoetigt
Xamarin.AndroidX.LocalBroadcastManager	`1.1.0.25`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Legacy.Support.Core.Utils benoetigt
Xamarin.AndroidX.Navigation.Common	`2.9.2.1`	MIT AND Apache-2.0	wird von Microsoft.Maui.Core, Xamarin.AndroidX.Navigation.Fragment, Xamarin.AndroidX.Navigation.Runtime benoetigt
Xamarin.AndroidX.Navigation.Common.Android	`2.9.2.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Navigation.Common benoetigt
Xamarin.AndroidX.Navigation.Fragment	`2.9.2.1`	MIT AND Apache-2.0	wird von Microsoft.Maui.Core benoetigt
Xamarin.AndroidX.Navigation.Runtime	`2.9.2.1`	MIT AND Apache-2.0	wird von Microsoft.Maui.Core, Xamarin.AndroidX.Navigation.Fragment, Xamarin.AndroidX.Navigation.UI benoetigt
Xamarin.AndroidX.Navigation.Runtime.Android	`2.9.2.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Navigation.Runtime benoetigt
Xamarin.AndroidX.Navigation.UI	`2.9.2.1`	MIT AND Apache-2.0	wird von Microsoft.Maui.Core benoetigt
Xamarin.AndroidX.NavigationEvent	`1.1.2.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity benoetigt
Xamarin.AndroidX.NavigationEvent.Android	`1.1.2.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.NavigationEvent benoetigt
Xamarin.AndroidX.Print	`1.1.0.4`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Legacy.Support.Core.Utils benoetigt
Xamarin.AndroidX.ProfileInstaller.ProfileInstaller	`1.4.1.8`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.ConstraintLayout benoetigt
Xamarin.AndroidX.RecyclerView	`1.4.0.3`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.ViewPager2, Xamarin.Google.Android.Material benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.AndroidX.ResourceInspection.Annotation	`1.0.1.22`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat, Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.SavedState	`1.5.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.AppCompat, Xamarin.AndroidX.Fragment benoetigt
Xamarin.AndroidX.SavedState.SavedState.Android	`1.5.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.ViewModelSavedState.Android, Xamarin.AndroidX.Navigation.Common.Android, Xamarin.AndroidX.SavedState benoetigt
Xamarin.AndroidX.Security.SecurityCrypto	`1.1.0.4-alpha07`	MIT AND Apache-2.0	wird von Microsoft.Maui.Essentials benoetigt
Xamarin.AndroidX.SlidingPaneLayout	`1.2.0.22`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Navigation.Fragment benoetigt
Xamarin.AndroidX.Startup.StartupRuntime	`1.2.0.8`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Emoji2, Xamarin.AndroidX.Lifecycle.Process, Xamarin.AndroidX.ProfileInstaller.ProfileInstaller benoetigt
Xamarin.AndroidX.SwipeRefreshLayout	`1.1.0.29`	MIT AND Apache-2.0	wird von Microsoft.Maui.Core benoetigt
Xamarin.AndroidX.Tracing.Tracing	`1.3.0.4`	MIT AND Apache-2.0	wird von Xamarin.Android.Glide, Xamarin.AndroidX.Activity, Xamarin.AndroidX.Camera.Camera2 benoetigt
Xamarin.AndroidX.Tracing.Tracing.Android	`1.3.0.4`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Tracing.Tracing benoetigt
Xamarin.AndroidX.Tracing.Tracing.Ktx	`1.3.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.Lifecycle benoetigt
Xamarin.AndroidX.Transition	`1.6.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Navigation.UI, Xamarin.AndroidX.SlidingPaneLayout, Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.VectorDrawable	`1.2.0.8`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.AppCompat.AppCompatResources, Xamarin.AndroidX.VectorDrawable.Animated, Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.VectorDrawable.Animated	`1.2.0.8`	MIT AND Apache-2.0	wird von Xamarin.Android.Glide, Xamarin.AndroidX.AppCompat.AppCompatResources benoetigt
Xamarin.AndroidX.VersionedParcelable	`1.2.1.6`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Core benoetigt
Xamarin.AndroidX.ViewPager	`1.1.0.7`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Fragment benoetigt
Xamarin.AndroidX.ViewPager2	`1.1.0.8`	MIT AND Apache-2.0	wird von Xamarin.Google.Android.Material benoetigt
Xamarin.AndroidX.Window	`1.4.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.SlidingPaneLayout benoetigt
Xamarin.AndroidX.Window.WindowCore	`1.4.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Window benoetigt
Xamarin.AndroidX.Window.WindowCore.Jvm	`1.4.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Window.WindowCore benoetigt
Xamarin.Build.Download	`0.11.4`	ungeklaert	wird von Xamarin.Firebase.Annotations, Xamarin.Firebase.Common, Xamarin.Firebase.Components benoetigt
Xamarin.Firebase.Annotations	`117.0.0.4`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Common, Xamarin.Firebase.Components, Xamarin.Firebase.Installations benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.Firebase.Common	`122.1.0.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Datatransport, Xamarin.Firebase.Installations, Xamarin.Firebase.Messaging benoetigt
Xamarin.Firebase.Components	`119.0.1.2`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Common, Xamarin.Firebase.Datatransport, Xamarin.Firebase.Installations benoetigt
Xamarin.Firebase.Datatransport	`120.0.1.3`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Messaging benoetigt
Xamarin.Firebase.Encoders	`117.0.0.26`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Encoders.JSON, Xamarin.Firebase.Encoders.Proto, Xamarin.Firebase.Messaging benoetigt
Xamarin.Firebase.Encoders.JSON	`118.0.1.18`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Messaging, Xamarin.Google.Android.DataTransport.TransportBackendCct benoetigt
Xamarin.Firebase.Encoders.Proto	`116.0.0.21`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Messaging, Xamarin.Google.Android.DataTransport.TransportRuntime benoetigt
Xamarin.Firebase.Iid.Interop	`117.1.0.26`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Messaging benoetigt
Xamarin.Firebase.Installations	`119.1.2`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Messaging benoetigt
Xamarin.Firebase.Installations.InterOp	`117.3.0.2`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Installations, Xamarin.Firebase.Messaging benoetigt
Xamarin.Firebase.Measurement.Connector	`120.0.1.14`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Messaging benoetigt
Xamarin.Google.Android.DataTransport.TransportApi	`4.1.1.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Datatransport, Xamarin.Firebase.Messaging, Xamarin.Google.Android.DataTransport.TransportBackendCct benoetigt
Xamarin.Google.Android.DataTransport.TransportBackendCct	`4.1.1.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Datatransport, Xamarin.Firebase.Messaging benoetigt
Xamarin.Google.Android.DataTransport.TransportRuntime	`4.1.1.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Datatransport, Xamarin.Firebase.Messaging, Xamarin.Google.Android.DataTransport.TransportBackendCct benoetigt
Xamarin.Google.Android.Material	`1.12.0.5`	MIT AND Apache-2.0	wird von Microsoft.Maui.Core, Xamarin.AndroidX.Navigation.UI benoetigt
Xamarin.Google.AutoValue.Annotations	`1.11.0.8`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.Camera2, Xamarin.AndroidX.Camera.Core, Xamarin.AndroidX.Camera.Lifecycle benoetigt
Xamarin.Google.Code.FindBugs.JSR305	`3.0.2.21`	MIT AND Apache-2.0	wird von Xamarin.Google.Crypto.Tink.Android benoetigt
Xamarin.Google.Crypto.Tink.Android	`1.18.0.1`	MIT AND Apache-2.0	wird von Microsoft.Maui.Essentials, Xamarin.AndroidX.Security.SecurityCrypto benoetigt
Xamarin.Google.ErrorProne.Annotations	`2.50.0.1`	MIT AND Apache-2.0	wird von GoogleGson, Xamarin.Firebase.Components, Xamarin.Firebase.Messaging benoetigt
Xamarin.Google.Guava.ListenableFuture	`1.0.0.32`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Browser, Xamarin.AndroidX.Camera.Camera2, Xamarin.AndroidX.Camera.Core benoetigt
Xamarin.GooglePlayServices.Base	`118.10.0.2`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Messaging, Xamarin.GooglePlayServices.CloudMessaging benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.GooglePlayServices.Basement	`118.10.0.2`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Common, Xamarin.Firebase.Iid.Interop, Xamarin.Firebase.Measurement.Connector benoetigt
Xamarin.GooglePlayServices.CloudMessaging	`117.4.0.1`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Messaging benoetigt
Xamarin.GooglePlayServices.Stats	`117.1.0.10`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Messaging benoetigt
Xamarin.GooglePlayServices.Tasks	`118.4.1.2`	siehe beigelegten Lizenztext	wird von Xamarin.Firebase.Common, Xamarin.Firebase.Iid.Interop, Xamarin.Firebase.Installations benoetigt
Xamarin.JavaX.Inject	`1.0.0.24`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Annotations, Xamarin.Google.Android.DataTransport.TransportRuntime benoetigt
Xamarin.Jetbrains.Annotations	`26.1.0.2`	MIT AND Apache-2.0	wird von Xamarin.Kotlin.StdLib, Xamarin.KotlinX.Coroutines.Core.Jvm benoetigt
Xamarin.JSpecify	`1.0.0.7`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Collection.Jvm, Xamarin.AndroidX.Concurrent.Futures, Xamarin.AndroidX.CoordinatorLayout benoetigt
Xamarin.Kotlin.StdLib	`2.4.0.1`	MIT AND Apache-2.0	wird von Square.OkIO.JVM, Xamarin.AndroidX.Activity, Xamarin.AndroidX.Activity.Ktx benoetigt
Xamarin.Kotlin.StdLib.Jdk7	`2.4.0.1`	MIT AND Apache-2.0	wird von Xamarin.Kotlin.StdLib.Jdk8 benoetigt
Xamarin.Kotlin.StdLib.Jdk8	`2.4.0.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Datatransport, Xamarin.Firebase.Encoders.JSON benoetigt
Xamarin.KotlinX.AtomicFU	`0.33.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Core, Xamarin.AndroidX.DataStore.Core.OkIO, Xamarin.AndroidX.Lifecycle.ViewModel benoetigt
Xamarin.KotlinX.AtomicFU.Jvm	`0.33.0.1`	MIT AND Apache-2.0	wird von Xamarin.KotlinX.AtomicFU benoetigt
Xamarin.KotlinX.Coroutines.Android	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Camera.Core, Xamarin.AndroidX.Camera.Lifecycle, Xamarin.AndroidX.Lifecycle.Runtime.Android benoetigt
Xamarin.KotlinX.Coroutines.Core	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Activity, Xamarin.AndroidX.Concurrent.Futures.Ktx, Xamarin.AndroidX.Core benoetigt
Xamarin.KotlinX.Coroutines.Core.Jvm	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Navigation.Runtime.Android, Xamarin.KotlinX.Coroutines.Android, Xamarin.KotlinX.Coroutines.Core benoetigt
Xamarin.KotlinX.Coroutines.Play.Services	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.Firebase.Common benoetigt
Xamarin.KotlinX.Serialization.Core	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences.Core, Xamarin.AndroidX.Lifecycle.ViewModelSavedState, Xamarin.AndroidX.Navigation.Fragment benoetigt
Xamarin.KotlinX.Serialization.Core.Jvm	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.Lifecycle.ViewModelSavedState.Android, Xamarin.AndroidX.Navigation.Common.Android, Xamarin.AndroidX.Navigation.Runtime.Android benoetigt
Xamarin.KotlinX.Serialization.Json	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Core.OkIO benoetigt
Xamarin.KotlinX.Serialization.Json.Jvm	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Core.OkIO.Jvm, Xamarin.KotlinX.Serialization.Json benoetigt

© 2026 Ordivis Platform · Christian Grams. Alle Rechte vorbehalten.

Name	Version	Lizenz	Verwendung
Xamarin.KotlinX.Serialization.Protobuf	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.AndroidX.DataStore.Preferences.Core benoetigt
Xamarin.KotlinX.Serialization.Protobuf.Jvm	`1.11.0.1`	MIT AND Apache-2.0	wird von Xamarin.KotlinX.Serialization.Protobuf benoetigt
ZXing.Net.Maui	`0.10.3`	MIT	wird von ZXing.Net.Maui.Controls benoetigt

FREMDINHALTE (KATALOGE) (3)

Der auf einer konkreten Anlage importierte Katalogstand steht im Systemstatus. Ein Katalog ohne ausgewiesenen Stand ist im Audit wertlos.

Name	Urheber	Lizenz	Herkunft
BSI IT-Grundschutz++ (Control-Layer)	Bundesamt fuer Sicherheit in der Informationstechnik	Werk des BSI, Weitergabe im Rahmen der Veroeffentlichung	github.com/BSI-Bund/Stand-der-Technik-Bibliothek, control_layer/Grundschutz++/Grundschutz++-resolved_catalog.json
ISO/IEC 27001:2022 Annex A (Massnahmenverzeichnis)	Abbildung aus der BSI-Bund/Stand-der-Technik-Bibliothek	Werk des BSI, Weitergabe im Rahmen der Veroeffentlichung; der Normtext selbst ist NICHT enthalten	OSCAL-Katalog derselben Bibliothek
Prozesskatalog der oeffentlichen Verwaltung	Grams IT, auf Grundlage oeffentlich zugaenglicher Prozesslandkarten	Bestandteil der Ordivis Platform	mitgeliefert als Anfangsbestand (PublicSectorCatalogSeeder)

Verwandte Themen

- **ID-00 – Handbuch-Übersicht & Impressum** – Dokumentenkarte und Rollenübersicht
- **ID-44 – Rollen- & Rechte-Matrix** – der vollständige Rechtekatalog
- **ID-01 – Einführung & Grundlagen** – Grundlagen und Begriffe im Zusammenhang

© 2026 Christian Grams / Ordivis Platform. Alle Rechte vorbehalten.