

DATENBLATT

EULANDA X-Suite



EULANDAXREST 1.0.5 · EULANDAXCLIENT 1.1.3 · STAND 25.09.2026

Die neue Zugangsschicht auf die EULANDA-Warenwirtschaft: eine dokumentierte REST-API, ein MCP-Server für KI-Werkzeuge und eine Weboberfläche mit eingebautem Assistenten für Browser, Tablet und Desktop. Auf derselben Datenbank wie EULANDA Klassik.

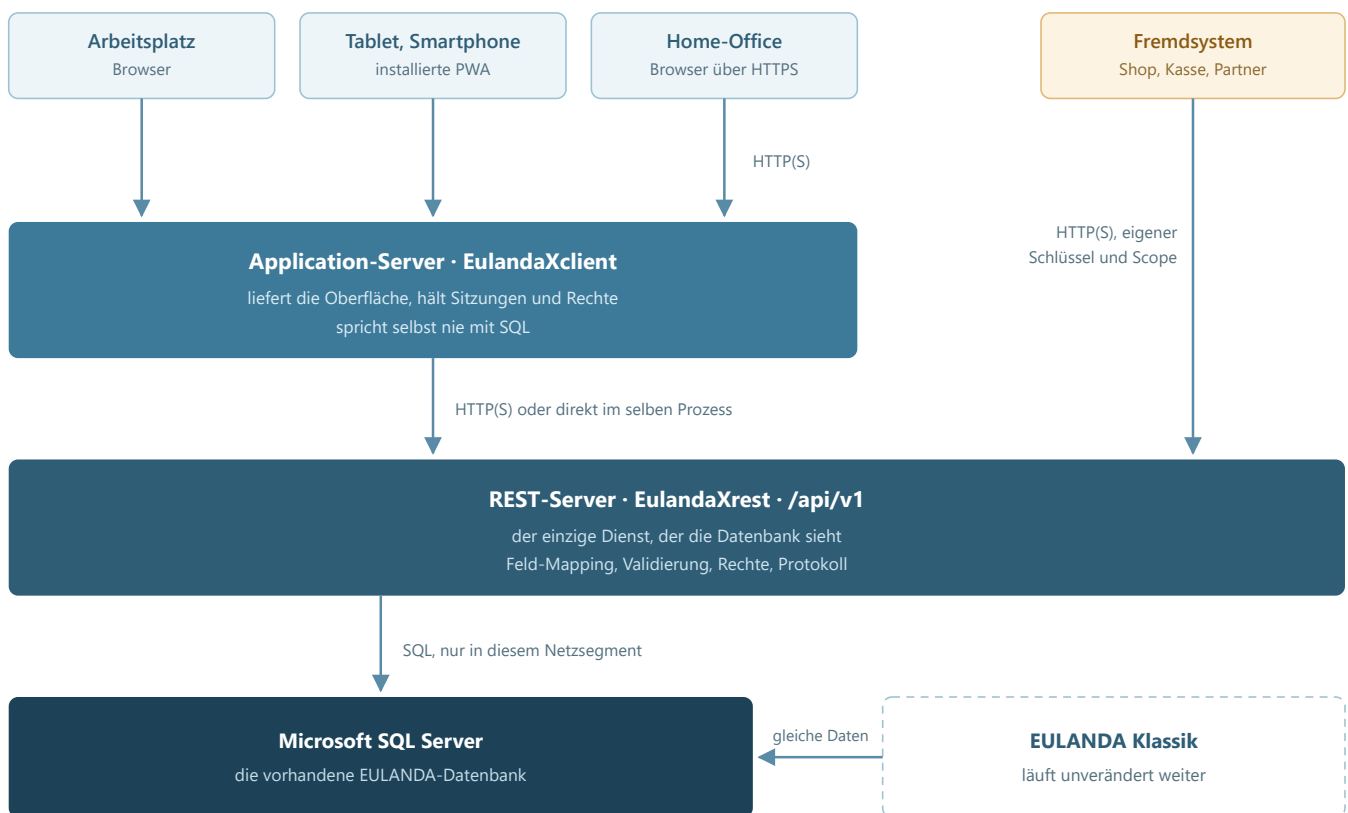
Worum es geht. EulandaX ersetzt den direkten SQL-Zugriff jedes Arbeitsplatzes durch genau einen Dienst. Alles andere spricht nur noch HTTP(S) - Browser im Haus, Tablet im Lager, Filiale ohne VPN, Onlineshop von außen. Die Daten bleiben dabei exakt die der bestehenden EULANDA-Installation, es gibt keine zweite Datenbank und keinen Abgleich.

- **Nichts wird neu erfunden** - die kaufmännische Logik rechnet seit jeher der SQL-Server. Die API ruft auf, was seit Jahren im Einsatz ist.
- **Keine Installation am Arbeitsplatz** - ein Browser genügt, auf Tablets auch als installierbare App (PWA).
- **Kein VPN und kein SQL-Zugang im Netz** - nur der REST-Server sieht die Datenbank.
- **Parallelbetrieb mit EULANDA Klassik** - dieselbe Datenbank, dieselben Buchungswege, Ablösung Schritt für Schritt.
- **Wächst mit** - vom Einzelplatz bis zu mehreren Servern hinter einem Load-Balancer, ohne Umbau der Anwendung.
- **Erweiterungen ohne Fork** - eigene Fachmodule kommen als Plugin dazu, mitsamt Menüpunkten, Masken und Rechten. Der Standard bleibt dabei unberührt und updatefähig.
- **REST, MCP und Assistent** - neben der dokumentierten REST-API bringt die Suite einen MCP-Server mit, über den KI-Werkzeuge mit den Daten arbeiten, und einen Assistenten direkt in der Oberfläche. Er erledigt Aufgaben wie „Lege mir ein Angebot für Kunde Mustermann mit den Artikeln A, B und C an.“
- **Im Echtbetrieb** - seit September 2026 bei EULANDA selbst im Tagesgeschäft und bei den ersten Kunden eingerichtet. Die Schnittstelle ist vollständig dokumentiert unter sdk.eulanda.eu/rest.

Für die Entscheidung in fünf Sätzen. Es gibt keinen Umstellungstag: Beide Welten arbeiten gleichzeitig auf denselben Daten, und jede Abteilung wechselt dann, wenn ihre Maske fertig ist. Es gibt nichts zu verteilen: Wer heute im Browser arbeiten soll, bekommt eine Adresse, kein Setup. Es entstehen keine zweiten Rechenwege, weil die kaufmännische Logik dort bleibt, wo sie seit Jahren liegt - in der Datenbank. Eigene Anpassungen bleiben eigene Module und werden von Updates nicht überfahren. Und der Betrieb wird einfacher statt komplizierter, weil der Datenbank-Port das Serverraum nicht mehr verlässt.

1. Das Prinzip

Klassisch braucht jeder Arbeitsplatz eine eigene SQL-Verbindung: Treiber, UDL-Datei, offener Datenbank-Port, für den Außendienst eine VPN-Brücke. In EulandaX gibt es **genau einen Dienst mit Datenbankzugriff**, den REST-Server. Er stellt die Daten als Endpunkte bereit (zum Beispiel `/api/v1/addresses`). Jeder Konsument holt sich dort JSON, ändert es und schickt es zurück.



Vier Schichten, ein Datenbestand: nur der REST-Server sieht SQL, alles darüber spricht HTTP(S).

Die zwei Server-Rollen

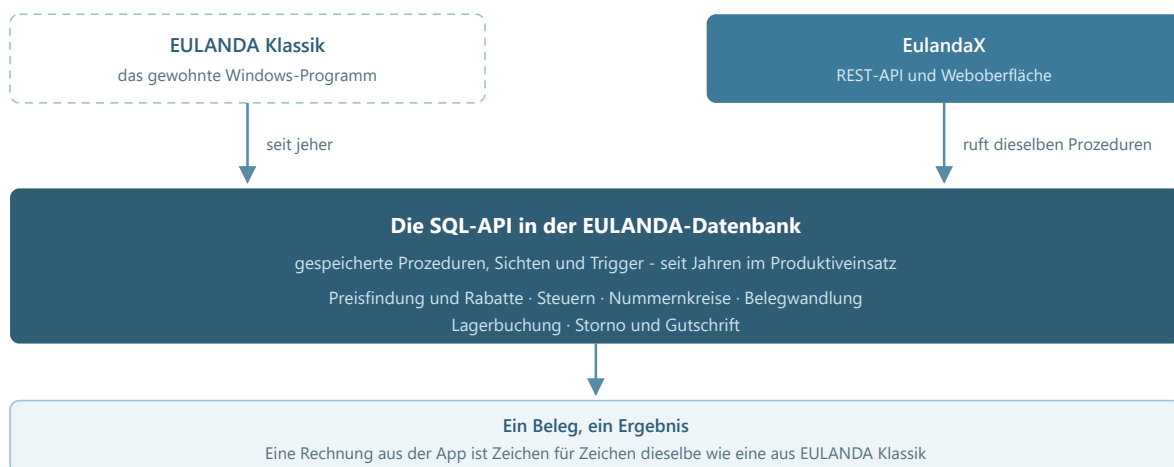
	EulandaXrest (REST-Server)	EulandaXclient (Application-Server)
Aufgabe	Daten-API /api/v1/*	liefert die HTML-Oberfläche und bedient sie
Spricht mit	SQL Server	ausschließlich mit EulandaXrest
Zustand	zustandslos, jeder Aufruf trägt seine Anmeldung selbst	hält Geräte- und Anmeldesitzungen
Steht	auf dem SQL-Rechner oder in dessen LAN	überall mit HTTPS-Sicht auf den REST-Server
Standard-Port	8100	8102

Der Browser spricht **nie** direkt mit dem REST-Server; den REST-Aufruf macht der Application-Server serverseitig. Damit bleibt der API-Schlüssel auf dem Server und verlässt nie das Endgerät. Stehen beide Dienste auf einer Maschine, läuft der REST-Teil **im selben Prozess** ab: derselbe Code, derselbe Vertrag, nur ohne Netzwerkweg. Dieselbe Oberfläche funktioniert deshalb ohne eine einzige Codeänderung vom Einzelplatz bis zum Mehrstandort-Betrieb.

2. Nichts wird neu erfunden

Das ist der Grund, warum dieses Projekt kein Jahrzehnt braucht: **die kaufmännische Logik von EULANDA liegt seit jeher in der Datenbank**, nicht im Windows-Programm. Preise, Rabatte und Staffeln, Steuern, Nummernkreise, die Wandlung vom Angebot über Auftrag und Lieferschein zur Rechnung, Lagerbuchungen, Storno und Gutschrift - all das erledigen gespeicherte Prozeduren und Sichten im SQL Server. EULANDA Klassik ist im Kern eine Oberfläche, die diese SQL-Schnittstelle bedient.

Der REST-Server rechnet deshalb nichts selbst. Er prüft die Anfrage, übersetzt die Feldnamen und ruft anschließend genau dieselbe Prozedur auf, die auch das Windows-Programm aufruft. Es gibt keinen zweiten Rechenweg, der irgendwann auseinanderlaufen könnte.



Beide Oberflächen bedienen dieselbe SQL-Schnittstelle. Die API ist ein neuer Zugang, kein Nachbau.

Was das praktisch bedeutet:

- **Keine Doppelpflege von Geschäftsregeln.** Ändert sich eine Regel in der Warenwirtschaft, gilt sie im neuen Client sofort mit - ohne dass jemand sie ein zweites Mal programmiert.
- **Kein Prüfaufwand für die Rechenwege.** Getestet wird der Zugang, nicht die Kaufmannslogik. Die ist dieselbe, die täglich in tausenden Belegen läuft.
- **Keine Abweichungen im Belegbild.** Nummernkreise, Rundungen und Steuerausweis kommen aus derselben Quelle - es kann gar nicht zu zwei Wahrheiten kommen.
- **Der Ausbau geht schnell.** Eine neue Ressource in der API ist im Wesentlichen die Zuordnung englischer Feldnamen auf vorhandene Sichten und Prozeduren, nicht die Entwicklung neuer Fachlogik.

Deshalb ist die Datenschicht bereits fertig. Die REST-API deckt den Funktionsumfang der Warenwirtschaft schon heute ab: Adressen und Kontakte, Artikel mit Preisen, Beständen, Seriennummern und Stücklisten, Merkmale, Lieferanten und Einkauf, die Belegkette vom Angebot über Auftrag und Lieferschein bis zur Rechnung samt Storno und Gutschrift, Lager, Kasse mit TSE, Zahlungen, DMS, Statistik, Nummernkreise, Firmenstamm und Registry. Alles davon ist dokumentiert und lässt sich direkt ausprobieren:

sdk.eulanda.eu/rest. Was jetzt Schritt für Schritt entsteht, sind die **visuellen Hüllen** darüber - die Masken der Weboberfläche.

3. Die Oberfläche

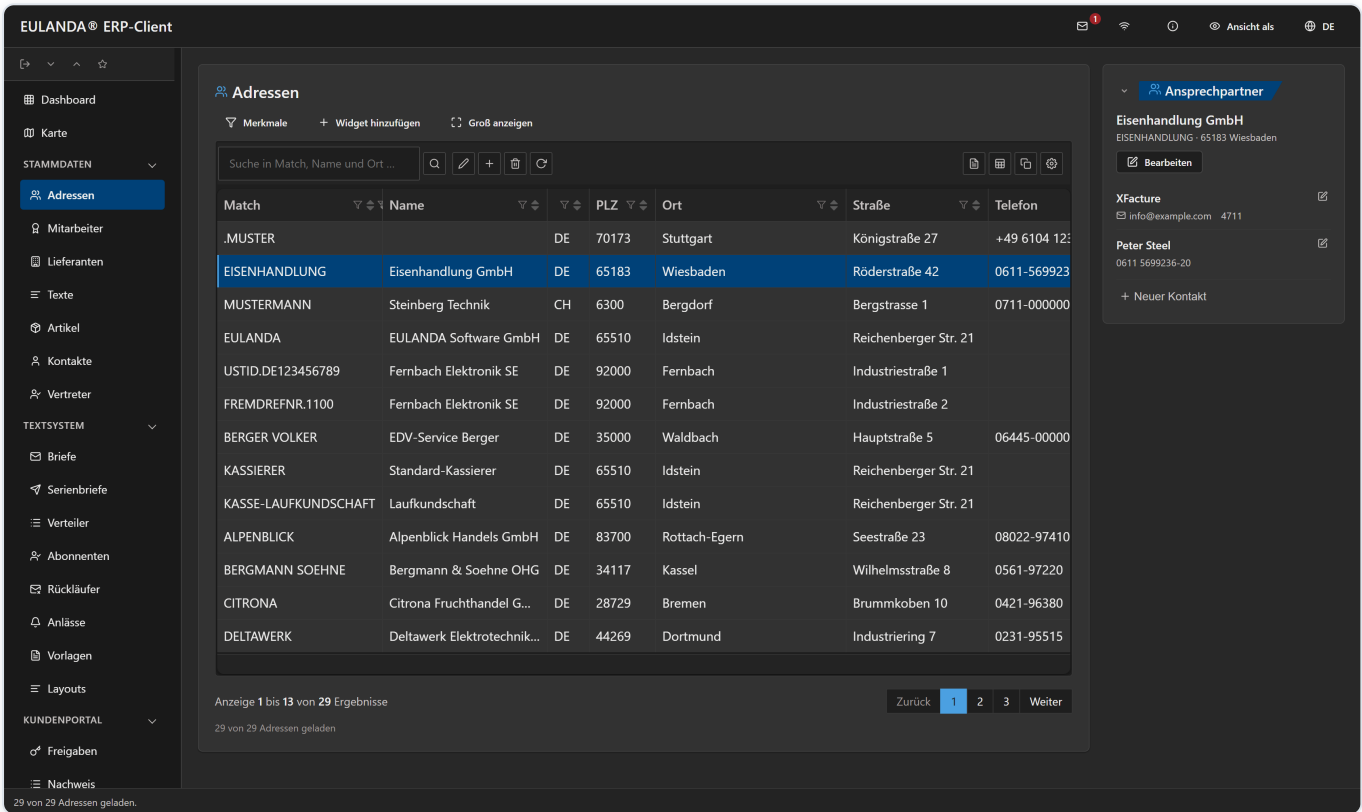
Eine einzige HTML-Anwendung bedient alle Betriebsarten. Sie wird einmal auf dem Application-Server aktualisiert und ist damit sofort auf jedem Gerät aktuell - kein Rollout, keine Paketverteilung.

Desktop-Fenster
Startet wie ein Windows-Programm (WebView2), holt die Daten über denselben Vertrag.

Browser im Netz
Adresse aufrufen, fertig. Null Installation, auch aus dem Home-Office.

PWA auf dem Tablet
Symbol auf dem Startbildschirm, eigenes Fenster. Android, iPad, Windows-Tablet.

Fremdsystem
Shop, Kasse oder Partnerprogramm sprechen direkt die REST-API mit eigenem Schlüssel.



Adressliste im Browser: Suche, Merkmalfilter, Ansprechpartner-Panel, Rollenumschalter "Ansicht als"

EULANDA® ERP-Client

Ansicht als DE

Dashboard
Karte
STAMMDATEN
Adressen
Mitarbeiter
Lieferanten
Texte
Artikel
Kontakte
Vertreter
TEXTSYSTEM
Briefe
Serienbriefe
Verteiler
Abonnenten
Rückläufer
Anlässe
Vorlagen
Layouts
KUNDENPORTAL
Freigaben
Nachweis

29 von 29 Adressen geladen.

Adresse: EISENHANDLUNG

Satz 2 von 29

Zurück Speichern Löschen Einstellungen Aktionen

Allgemein Erweitert

Kennzeichnung

Match EISENHANDLUNG
Eindeutiges Kürzel der Adresse (Pflichtfeld bei Neuanlage).

Fibu-Konto 10200

Gruppe

Fremde Referenz 123456789-4711

Fremde Nummer 54321

Anschrift

GLN 4317784008914

Name Eisenhandlung GmbH

Name 2 Hebebühnen aus ALU

Name 3 Industriefläche Süd

Straße Röderstraße 42

Ort D 65183 Wiesbaden

Kommunikation

Telefon 0611-5699236-14

Mobil

Fax

E-Mail einkauf@eisenhandlung.de

Webseite www.wiesbaden.de

Sprache Deutsch

Zuordnung

Vertreter 4

Provision (%)

Konditionen

Anrede

Übersicht

Eisenhandlung GmbH / Hebebühnen aus ALU / Industriefläche Süd

Match EISENHANDLUNG

Straße Röderstraße 42

Ort DE 65183 Wiesbaden

Telefon 0611-5699236-14

E-Mail einkauf@eisenhandlung.de

USt-IdNr. DE20215869

Merkmale

- Daten Wartung
- Gesperrt
- Regionen
- Shop
- TEST

Bild

Die Adressmaske mit Feldhilfe, Merkmalbaum, Karteikarte und Bild - Felder und Sichtbarkeit hängen an der Rolle

Anmelden

Bitte mit Benutzername und Passwort anmelden.

Benutzer

Passwort

Anmelden

Anmeldung mit Benutzer und Passwort - hier im hellen Design

Hell oder dunkel ist eine Sache von einem Klick und gilt je Gerät. In der Voreinstellung folgt die Anwendung dem, was das Betriebssystem vorgibt. Die übrigen Abbildungen in diesem Datenblatt zeigen das dunkle Design.

EULA...

Ansicht als DE

Adressen

Merkmale Groß anzeigen

Suche in Match, Name und Ort ...

Anzeige 1 bis ... Zurück 1 2 3 Weiter

.MUSTER
10200
DE · 70173 · Stuttgart · Königstraße 27 · +49 6104 12345

EISENHANDLUNG
10200
Eisenhandlung GmbH · DE · 65183 · Wiesbaden · Röderstraße 42 · 0611-5699236-14

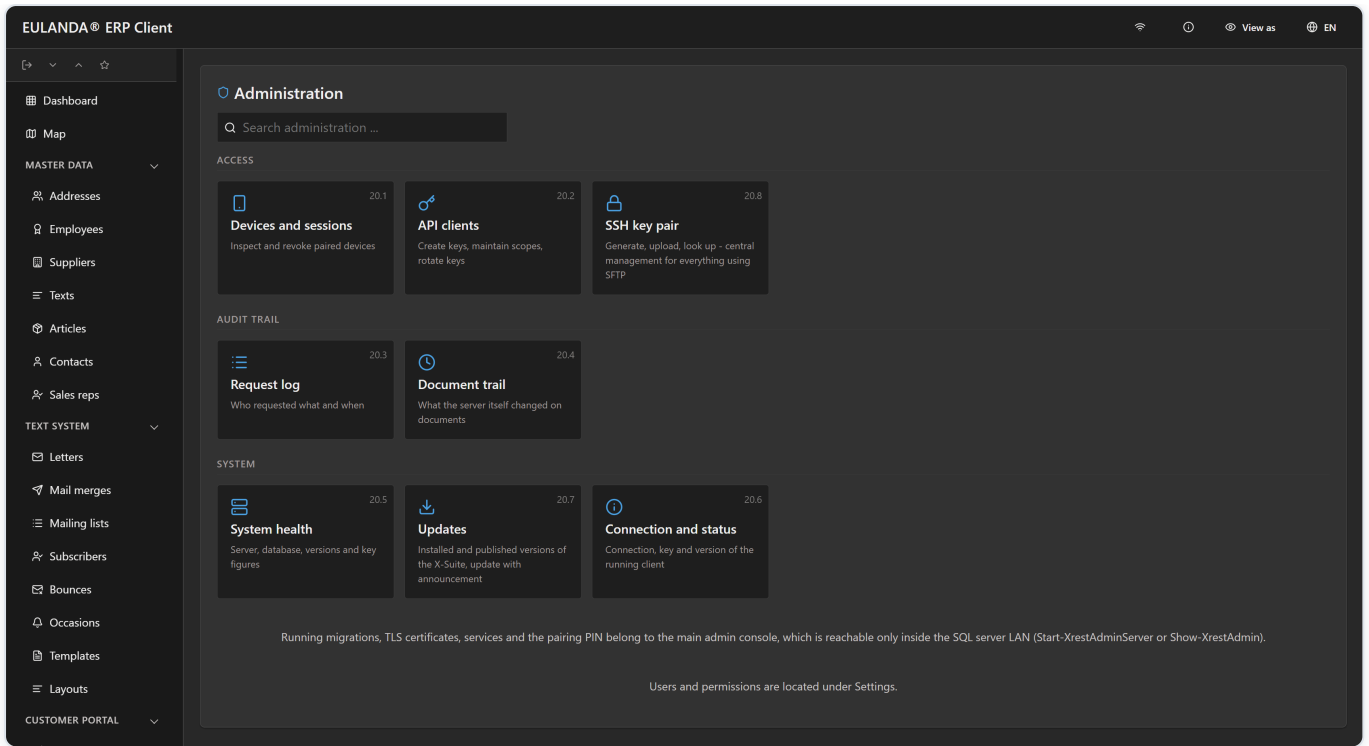
MUSTERMANN
10001
Steinberg Technik · CH · 6300 · Bergdorf · Bergstrasse 1 · 0711-000000

EULANDA
10035
EULANDA Software GmbH · DE · 65510 · Idstein · Reichenberger Str. 21

USTID.DE123456789
10034
Fernbach Elektronik SE · DE · 92000 · Fernbach · Industriestraße 1

29 von 29 Adressen geladen.

Dieselbe Anwendung auf dem Smartphone: die Tabelle wird automatisch zur Kartenliste



Verwaltung im Client: gekoppelte Geräte, API-Schlüssel mit Scopes, Zugriffs-Log und Systemzustand. Diese Abbildung zeigt die englische Oberfläche - umgeschaltet wird im laufenden Betrieb

4. Skalierbarkeit

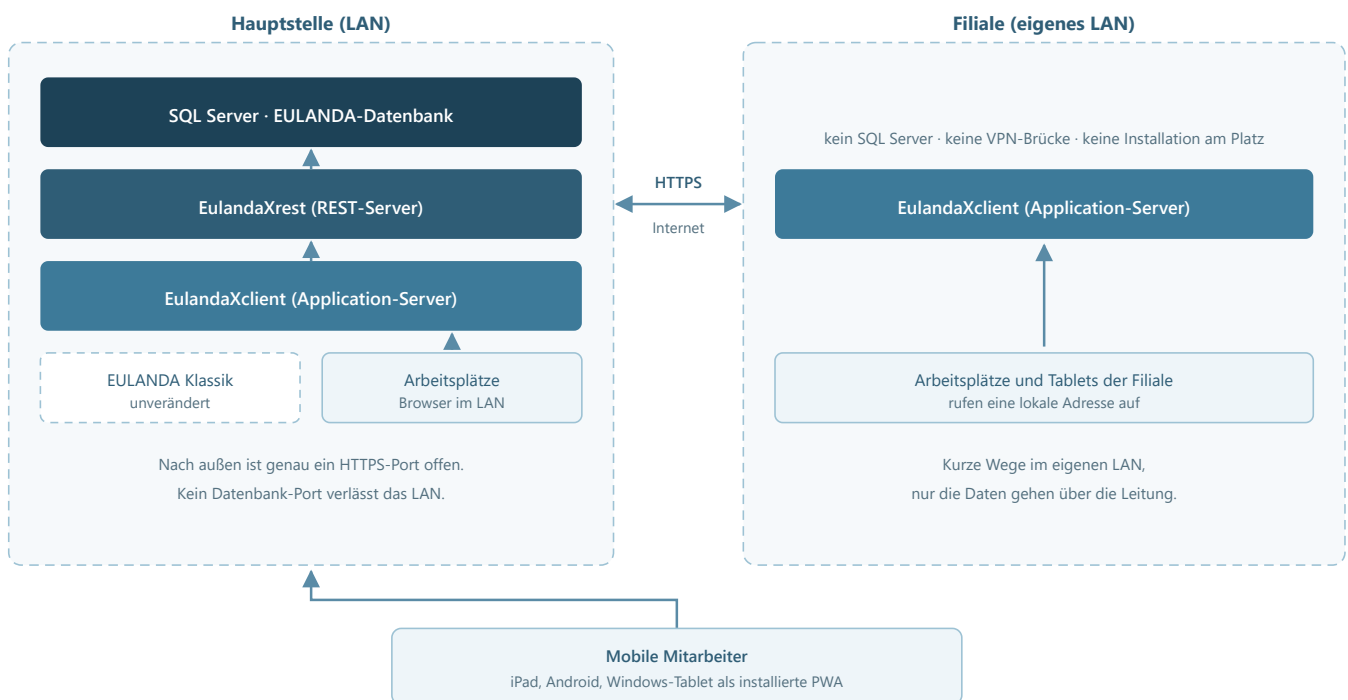
Parallele Bearbeitung. Beide Server beantworten mehrere gleichzeitige Anfragen echt nebenläufig; eine langsame Abfrage blockiert die anderen nicht. Jede Anfrage läuft dabei über eine eigene, auf Lebendigkeit geprüfte Datenbankverbindung - kein Verbindungsaufbau pro Klick. Der Parallelbetrieb ist ein bewusster Schalter, damit eine kleine Installation schlank und vorhersagbar bleibt. Gemessen im Testbetrieb: 24 Anfragen nacheinander 2818 ms, dieselben 24 parallel 927 ms.

Horizontal wachsen. Der REST-Server ist zustandslos. Reicht die Leistung für viele Browser und Tablets nicht mehr, stellt man weitere REST-Server auf dieselbe Datenbank und verteilt die Anfragen über einen Balancer. Auch die Oberflächen-Schicht lässt sich verteilen, typischerweise ein Application-Server je Standort.

Sitzungen überleben Neustarts. Kopplung und Anmeldung sind in der Mandanten-Datenbank festgeschrieben (Kopplung 180 Tage, Anmeldung 12 Stunden, jeweils gleitend). Ein Server-Update kostet niemanden eine neue PIN-Eingabe und keine neue Anmeldung.

Größenordnung. Eine Maschine bedient rund 50 Benutzer je Mandant komfortabel. Darüber hinaus wird nicht umgebaut, sondern hinzugestellt - die Anwendung selbst bleibt unverändert.

Mehrere Standorte, ohne VPN und ohne zweiten SQL-Server



Die Filiale braucht weder Datenbank noch VPN. Ihr Application-Server spricht über HTTPS mit dem REST-Server der Hauptstelle.

Wer es noch einfacher möchte, lässt den lokalen Server in der Filiale weg und ruft direkt die Adresse der Hauptstelle im Browser auf. Beide Wege liefern dieselbe Anwendung; der lokale Application-Server verkürzt lediglich die Wege im Filial-LAN.

5. Sicherheit

- **Der API-Schlüssel ist die Vertrauensgrenze.** Er wird nur als Hash gespeichert (PBKDF2-SHA256) und einmalig bei der Anlage angezeigt.
- **Scopes begrenzen die Routen.** Ein Schlüssel mit `addresses:read` liest Adressen und sonst nichts.
- **Rolle je Schlüssel.** Kein Zugang ist automatisch Administrator; nur ausdrücklich vertrauenswürdige Schlüssel dürfen im Namen eines Benutzers handeln.
- **Benutzer, Rollen und Rechte** liegen als Daten in der Mandanten-Datenbank und werden serverseitig durchgesetzt, nicht im Browser. Felder, die eine Rolle nicht sehen darf, verlassen den Server nicht.
- **Geräte-Kopplung per PIN** im Serverbetrieb, mit Sperre gegen Durchprobieren.
- **Kein SQL vom Client.** Filter sind benannte Parameter auf einer Feld-Whitelist, jeder Wert wird parametrisiert übergeben.
- **TLS über http.sys**, koexistiert mit einem vorhandenen IIS auf Port 443. Kein IIS-Zwang, kein zusätzlicher Webserver.
- **Nachvollziehbarkeit:** jeder Schreibzugriff steht mit Client, Ressource, Schlüssel und Ergebnis im Zugriffs-Log.

6. Ein Datenbestand, kein Abgleich

Wie die Logik (Abschnitt 2) sind auch die Daten bereits da: EulandaX ist kein neues System mit eigener Datenhaltung, sondern ein neuer Zugang zur bestehenden.

- **Eine Datenbank.** Keine Zweitdatenbank, keine Synchronisation, kein Zeitversatz, keine Konflikte - und damit auch keine Schnittstelle, die nachts durchlaufen und morgens fehlen könnte.
- **Parallelbetrieb ohne Umstellungstag.** Die Buchhaltung arbeitet weiter im gewohnten Programm, Außendienst und Lager arbeiten schon im Browser - beide sehen im selben Moment dieselben Daten.
- **Bestehende Zusatzmodule laufen weiter.** Alles, was heute auf der Datenbank aufsetzt, sieht die über die API erfassten Belege genauso wie die aus dem Windows-Programm.
- **Kollisionsfrei.** Was EulandaX selbst braucht, liegt in einem eigenen Datenbankschema und wird über nummerierte, wiederholbare Migrationen eingespielt. Die vorhandene EULANDA-Update-Pipeline bleibt unberührt.

Ablösung statt Migration. Das neue System kann das alte Abteilung für Abteilung ersetzen - ohne Migrationsprojekt, ohne Stichtag und ohne Risiko, weil beide Welten jederzeit gleichzeitig auf denselben Daten arbeiten.

7. Erweitern ohne Fork

Kaum eine gewachsene Installation kommt ohne Anpassungen aus - eigene Felder, eigene Auswertungen, eine Verbandsanforderung, ein Ablauf, den es nur in diesem Haus gibt. Klassisch endet das oft in geänderten Standarddateien, und beim nächsten Update ist die Anpassung entweder weg oder das Update bleibt liegen. **EulandaX ist so gebaut, dass das nicht passieren muss.** Erweiterungen kommen als **Plugin** dazu und liegen vollständig neben dem Standard - der Kern wird nicht angefasst, nichts wird überschrieben.

Ein Plugin bekommt dafür auf jeder Ebene seinen eigenen Namensraum:

Ebene	Was dem Plugin gehört
Datenbank	ein eigenes Schema mit eigenen Migrationen. Kern-Tabellen werden nie verändert; auf Kern-Daten wird gelesen, geschrieben wird über die vorhandenen Kern-Routen.
REST	ein eigener Pfad <code>/api/v1/<plugin>/*</code> mit eigenen Scopes. Die Schnittstellen-Beschreibung des Plugins wird automatisch in die Dokumentation des Servers eingemischt.
Oberfläche	eine Beschreibung in JSON, aus der der Client Menüpunkte, Karteikarten, Tabellen, Formulare und Aktionen selbst baut - mit denselben Bausteinen wie der Standard.
Rechte	eigene Berechtigungen, die in die Rollen-Matrix einfließen. Menüpunkt und zugehörige Route hängen an derselben Berechtigung - unsichtbar im Client heißt auch abgewiesen am Server.
Sprache	eigene Übersetzungen im eigenen Schlüssel-Namensraum, die beim Seitenaufbau dazugemischt werden. Standardtexte zu überschreiben ist ausgeschlossen.

Weil alles im eigenen Namensraum liegt, ist der Lebenszyklus eine kurze Liste: **installieren** (Migrationen, Registrierung, Rechte), **deaktivieren** (Routen und Menü verschwinden, die Daten bleiben) und **deinstallieren** (Schema weg, Registrierung weg, Rechte weg) - rückstandsfrei und je Mandant einzeln. Ein Kern-Update kann eine Erweiterung nicht überschreiben, weil es nichts von ihr besitzt.

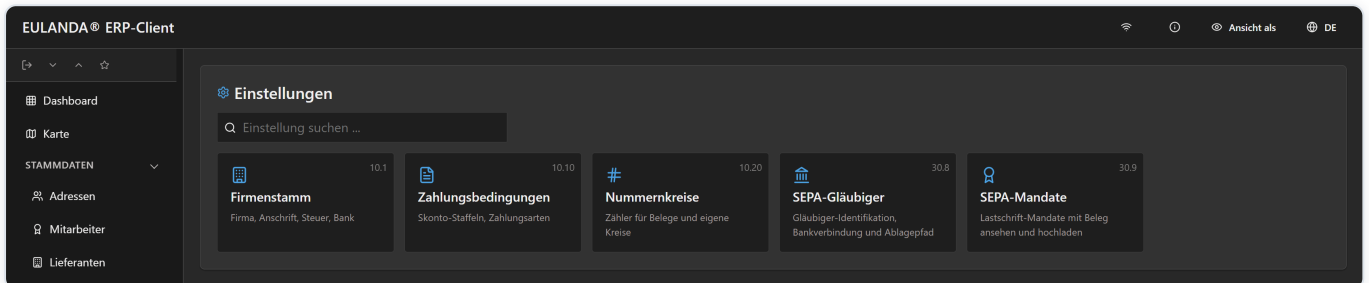
Die Oberfläche liefert das Plugin gleich mit

Der interessante Teil ist die Oberfläche: Ein Plugin muss dafür in aller Regel **keine einzige Zeile Anwendungs-JavaScript** schreiben. Es beschreibt, was es zeigen will, und der Client rendert es:

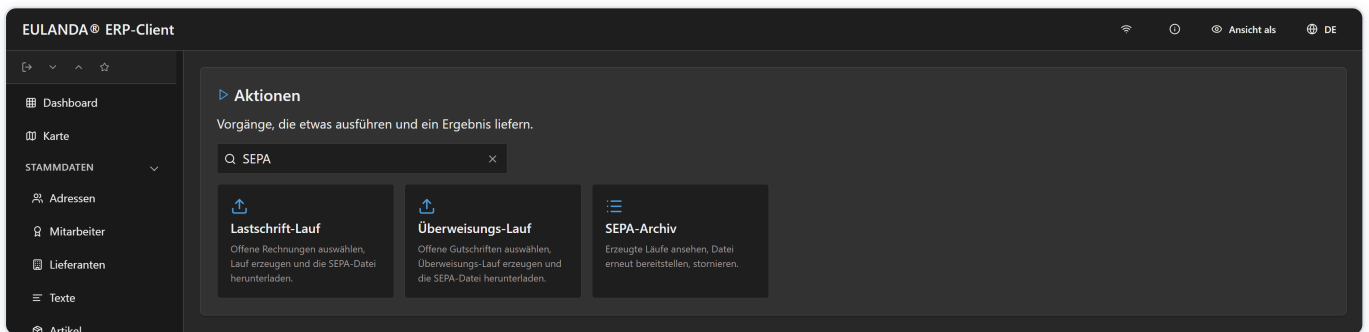
- **Eigene Menüpunkte** in der Hauptnavigation, mit Karteikarten darunter.
- **Tabellen mit Detailansicht** - Zeile anklicken, Datensatz ansehen, und am Datensatz hängen die passenden Aktionen (etwa erneut herunterladen oder stornieren).
- **Eigene Einstellungs-Kacheln**, die neben den Kacheln des Standards im Einstellungen-Bereich erscheinen.
- **Ein Aktionen-Bereich** für Dinge, die etwas tun statt etwas anzuzeigen: ausgelöste Läufe, Importe, Exporte - jeweils mit Ergebnis und Datei.
- **Dokumente aus dem Archiv** direkt am Datensatz: Beleg anzeigen, hochladen, entfernen - über dieselben Routen, die auch der Standard nutzt.

Für echte Sonderfälle (eine Karte, ein Spezial-Widget) gibt es eine definierte Einklinkstelle für ein eigenes Oberflächen-Modul. Sie ist die Ausnahme, nicht der Regelweg - und auch sie ist eine Ergänzung, kein Eingriff.

Beispiel aus der Praxis: SEPA. Das erste Fachmodul, das diesen Weg geht, ist die SEPA-Zahlungsabwicklung (derzeit im Pilotbetrieb): Mandate einsehen, Lastschrift- und Überweisungsläufe auslösen, das Archiv mit erneutem Download und Storno, Mandatsbelege aus dem Dokumenten-Archiv anzeigen, hochladen und löschen. Bemerkenswert daran ist, was nicht passiert ist: Die Fachlogik - Erzeugung der Datei, Verbuchung, Archivierung - ist derselbe Code, der vorher unter einer Windows-Oberfläche lief. Dazugekommen ist ein dünner Adapter auf die REST-Ebene und eine Beschreibung der Masken. Beide Oberflächen laufen währenddessen gleichzeitig auf denselben Tabellen.



Die Kacheln des Standards und die beiden des SEPA-Moduls in einer Reihe - dem Anwender fällt der Unterschied nicht auf, und genau das ist der Anspruch



Der Aktionen-Bereich für Vorgänge, die etwas ausführen und ein Ergebnis liefern. Die drei Einträge kommen vollständig aus dem Plugin, samt Beschreibung, Berechtigung und Ablauf

8. Mandanten, Lizenz und Betrieb

Mandanten. Der Normalfall ist der eigene Server im eigenen Haus mit einem Mandanten - so verhält sich das System ohne weitere Konfiguration. Für Hosting lässt sich Mehr-Mandanten-Betrieb zuschalten: je Kunde eine Subdomain und eine eigene Datenbank mit eigenem SQL-Login, sodass der SQL Server die Trennung selbst erzwingt. Geräte- und Anmelde-Token gelten immer nur unter ihrem Mandanten. Auch die aktiven Plugins hängen am jeweiligen Mandanten, nicht an der Installation.

Lizenz. Die Lizenz ist ein signierter Block, der an die Firmendaten des Mandanten gebunden ist. Er trägt die Zahl der Vollarbeitsplätze (gleichzeitig angemeldete Benutzer) und die freigeschalteten Zusatzprodukte. Daneben gibt es **App-Plätze** für zweckgebundene Mini-Clients mit eigener Platzzahl - eine Stechuhr für 500 Fahrer braucht damit keine 500 Vollarbeitsplätze.

Auslaufen sperrt niemanden aus. Fehlt oder endet die Lizenz, schaltet der Server auf **Lesebetrieb**: Alle Daten bleiben sichtbar, nur das wertverändernde Buchen ist gesperrt - und Anmeldung, Verwaltung und das Einspielen einer neuen Lizenz bleiben offen. Zusatzmodule, die ein Kunde in der klassischen Welt bereits lizenziert hat, werden beim Ausstellen der neuen Lizenz automatisch übernommen; er kauft nichts zweimal.

Verwaltung. Beide Server bringen eine Weboberfläche für den Betrieb mit: Erstlauf-Assistent und Einrichtungs-Diagnose, Benutzer, Rollen und Geräte, API-Clients mit Scopes und Schlüssel-Wechsel, Migrationen, Lizenzstand - und eine Auswertung des Zugriffs-Logs mit Kennzahlen, Verlauf und den häufigsten Clients und Routen. Fernwartung läuft über dieselben Routen wie alles andere, es braucht also keine Sitzung auf dem Server.

9. Stand heute und Perspektive

Werdegang	Begonnen im Januar 2025 mit den Grundfunktionen, damals als EulandaXtools. Ab Februar 2026 kamen die Grundlagen für die Oberfläche in HTML, das WebKit und das Modulsystem; darauf sind seitdem die Fachmodule entstanden, bis heute.
Im Echtbetrieb	Seit September 2026 bei EULANDA selbst im Tagesgeschäft und bei den ersten Kunden eingerichtet - direkt auf der EULANDA-Datenbank, ohne Abgleich und ohne Doppelpflege.
Oberfläche	Adressen, Artikel, Angebote, Aufträge, Lieferscheine, Rechnungen und Bestellungen im Browser, auf Tablet und Handy, auch als App auf dem Startbildschirm. Auswertungen als Kacheln, die jeder selbst zusammenstellt. Benutzer, Rollen und Rechte je Mandant.
Module	Kundenportal (Belege, Verträge und Anfragen ohne Kennwort), Ticketsystem mit Zeiterfassung, Telefonie mit 3CX, Serviceverträge mit wiederkehrender Rechnung, Briefe, Serienmails und Newsletter, E-Rechnung im Ausgang (ZUGFeRD, XRechnung), EDI mit Handelspartnern und SEPA-Zahlungsverkehr.
KI und Schnittstellen	Die REST-API für jedes Fremdsystem, ein MCP-Server (Model Context Protocol), über den KI-Werkzeuge mit den Daten der Suite arbeiten, und ein Assistent direkt in der Oberfläche: „Lege mir ein Angebot für Kunde Mustermann mit den Artikeln A, B und C an“ oder „Stell mir die 50 besten Artikel der Warengruppe XY und die Top-Kunden als Excel-Datei zusammen“. Er kennt den Aufbau der Suite und arbeitet direkt mit ihren Daten.
Einrichtung	Ein Setup richtet die Suite ohne Nachfragen ein - neu, neben einer bestehenden EULANDA oder als Update. Es prüft vorher Speicher, Netz, Firewall und Router, installiert bei Bedarf den SQL Server und gibt jeder Firma eine eigene Adresse mit gültigem Zertifikat. Updates holt die Suite danach selbst.
Als Nächstes	Eingangsrechnungen mit Vorkontierung und Übergabe an DATEV, danach Lager und Inventur, Mahnwesen sowie Projekt-, Abschlags- und Schlussrechnung - bis die Suite alles kann, was die Klassik kann.
Betrieb	Parallelbetrieb mit der Klassik, Plugin-System, Mehr-Mandanten-Betrieb, dauerhafte Sitzungen, Lizenzierung und die Auslieferung ohne Handarbeit sind da. Offen: Lastverteilung über mehrere Server.

Der Schnittstellenvertrag ist dabei auf Bruchfreiheit ausgelegt: Felder werden nur ergänzt, nie umbenannt oder entfernt, es gibt keine neuen Pflichtfelder für bestehende Clients, und Schema-Änderungen kommen ausschließlich als neue, wiederholbare Migration. Ein älterer Application-Server läuft gegen einen neueren REST-Server weiter - Updates lassen sich also nacheinander einspielen, ohne einen Stichtag für alle Standorte.

10. Technisches Kurzprofil

Datenbank	Microsoft SQL Server mit der vorhandenen EULANDA-Datenbank
Server	Windows, PowerShell 5.1 oder neuer, http.sys (HttpListener)
Fremdprodukte	keine - kein IIS-Zwang, kein Kestrel, kein Paketwald
Protokoll	REST/JSON, versioniert unter /api/v1 , Fehler als problem+json mit stabilem Code
KI	MCP-Server (Model Context Protocol) für KI-Werkzeuge und ein Assistent in der Oberfläche
Anmeldung	API-Schlüssel als Bearer-Token, Scopes je Client, Benutzerrollen je Mandant
Endgeräte	jeder aktuelle Browser, Android, iPad, Windows-Tablet - keine Installation nötig
PWA	Installation verlangt HTTPS (oder localhost)
Sprachen	Oberfläche und API-Fehlertexte deutsch und englisch, je Anwender umschaltbar
Erweiterungen	Plugins mit eigenem Schema, eigenen Routen, eigenen Rechten und deklarativer Oberfläche - je Mandant installierbar
Einrichtung	ein Setup für Server und Datenbank, eigene Adresse mit Zertifikat (etwa firma.x.eulanda.eu), Updates aus der Suite
Lizenz	signierter Block je Mandant, Vollarbeitsplätze und App-Plätze, Auslaufen schaltet auf Lesebetrieb

EULANDA[®] ist eine eingetragene Marke der EULANDA Software GmbH. Die Abbildungen zeigen die Oberfläche mit Testdaten.
Vollständige Dokumentation der REST-API: sdk.eulanda.eu/rest

[EulandaXrest 1.0.5](#) · [EulandaXclient 1.1.3](#) · [Stand 25.09.2026](#)