

D-ESS Benutzerhandbuch (under construction)

- [D-ESS Benutzerhandbuch für Administratoren \(historisch\) \(#page-249\)](#)
- [DESS · DB komplett zurücksetzen \(#page-266\)](#)

D-ESS Benutzerhandbuch für Administratoren (historisch)

D-ESS Benutzerhandbuch für Administratoren

Status (Stand 2026-05-23): Dieses Dokument wurde 2021 von **Ralf Bauridl (bundesgärten.at)** als Pilotbeitrag begonnen und ist seitdem **nicht weitergeführt** worden. Es spiegelt den Stand der Pilotphase wider und ist als **historischer Schnappschuss** zu lesen, nicht als aktuelles Handbuch.

Aktuelle Doku-Quellen:

- Code + Stammdatenmodell: GitLab [dess/dess](#)
- Integrations-Konzept BAB + OpenProject: KIROOT-Vault [wiki/concepts/integration-dess-openproject.md](#)
- SSO-Konfiguration + Incident-Historie: KIROOT-Vault [wiki/incidents/2026/](#)

Für Endanwender:innen-Schulungen ist Stand 2026-05 keine aktuelle Doku-Quelle vorhanden — als Backlog-Punkt erfasst (siehe „Offene Punkte“ am Seitenende).

1. Ziel + Prinzipien (2021)

Ursprünglich verfasst als kollaboratives Wiki-Handbuch für die DESS-Zeiterfassung der Dienststellen. Leitprinzipien:

- **KISS** — Keep It Simple, Stupid: schlanker Inhalt, schlankes Design.
- **WIKI-Prinzip** — jede:r Benutzer:in kann beitragen, ergänzen, korrigieren.

Beiträge wurden 2021 an ralf.bauridl@bundesgaerten.at geschickt. **Diese Sammel-Adresse ist Stand 2026-05 nicht mehr aktiv betreut.**

2. Funktionen von D-ESS

DESS ist die Arbeitszeiterfassung für Dienststellen zur Verrechnung von Arbeitskosten auf interne und externe Leistungen:

- Erfassung + Dokumentation der An- und Abwesenheitszeiten
- Erfassung + Dokumentation der geleisteten Tagesarbeitszeiten
- Zuordnung + Dokumentation der Leistungszeit zu Projekten und Leistungsteilprozessen (intern/extern)
- Auswertung + Interpretation der erfassten Leistungsstunden
- Verrechnung von Leistungskosten (Beamte, VB, KV, Leiharbeitsverträge)
- Feststellung von Verrechnungssätzen (tatsächliche Arbeitskosten der Dienststelle nach Verrechnungsgruppen)
- Erfassung + Vorbereitung der Verrechnung von Mehrdienstleistungen (MDL)
- Grundlage für Statistiken
- Protokollierung der Erbringung gesetzlicher Aufträge (z.B. Baumschutzgesetz, Schneeräumung)
- Strategische Entscheidungsgrundlage

3. Erster Login

Stand 2026-05: Login erfolgt für BAB-Benutzer:innen ausschließlich per **Keycloak-SSO** (Button „Login mit SSO“). Lokale Passwort-Logins (passport-local mit MD5-Hash) sind nur noch für Notfall-Accounts vorgesehen.

Aufruf der Zeiterfassung:

Dienststelle	URL
BAB	https://dess.agrarforschung.at (https://dess.agrarforschung.at)

SSO-Login:

1. URL aufrufen
2. „Login mit SSO“ anklicken
3. Keycloak-Login mit AD-Account (BAB\ bzw. UPN)
4. Weiterleitung zurück nach DESS

Lokales Admin-Login (Bootstrap / Notfall):

1. Benutzername + Passwort vom Hauptadmin (Thomas Skidmore oder Roland Neissl) anfordern.
2. Nach erstem Login Passwort selbst ändern.

4. Auswahlmenü + Stammdaten

Über die Menüleiste werden die Stammdatengruppen aufgerufen:

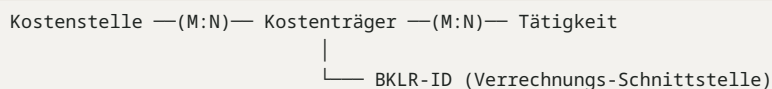
- **Benutzerverwaltung** — User, Benutzergruppen, AZM, Stehende Buchungen, MDL
- **Stammdaten** — Abteilungen, Kostenstellen, Kostenträger, Kostenträgergruppen, Tätigkeiten, KS↔KT- und KT↔Tätigkeit-Verknüpfungen
- **Zeiterfassung** — Eigene Buchungen, Jahresübersicht, Gruppenbuchung
- **Berichte** — Standard-Reports (Dienstzeitnachweis, KLR-Export, ...) + SQL-Query-Berichte
- **Quartalsabschluss** — MDL-Pflege
- **System** — Berechtigungen, Konfiguration, Stundenarten

5. Benutzer:in anlegen (Admin-Workflow)

1. **Benutzerverwaltung** → **Benutzer** → **Neu**
2. Pflichtfelder ausfüllen:
 - Benutzername (Login)
 - Vor- + Nachname, E-Mail
 - Abteilung + Kostenstelle (für KLR-Zuordnung)
 - Authentifizierungstyp: `OIDC` für SSO-User, `AD` für reine AD-Auth, `DB` für lokale User
 - BKLR-ID (falls relevant)
3. **Benutzergruppe** zuweisen (definiert Default-AZM + Stehende Buchungen)
4. **Stehende Buchungen** ggf. user-spezifisch überschreiben
5. Speichern → User erscheint in der Login-Maske

6. Datenmodell-Übersicht

Die Stammdaten-Hierarchie ist **dreidimensional**:



Gültige Buchungstripel = Schnittmenge der M:N-Tabellen.

Volle Schema-Doku siehe Vault `wiki/concepts/integration-dess-openproject.md` Abschnitt 2.4.

7. Verwandte Themen

- **GitLab Repo** `dess/dess` (https://git.agrarforschung.at/dess/dess)
- KIROOT-Vault `wiki/concepts/integration-dess-openproject.md`
- KIROOT-Vault `wiki/incidents/2026/2026-05-21-dess-ss0-broken-after-kc26.md` (SSO-Login-Incident KC26)

Offene Punkte (Backlog)

- End-User-Handbuch** — kein gepflegtes Handbuch für reguläre Anwender:innen vorhanden; Bedarf prüfen.
- Screenshots aktualisieren** — alle Bildschirmfotos stammen aus der Pilotphase 2021 (DESS 0.x), aktuelle UI ist Angular 12+/1.0.14.
- Mit Thomas Skidmore klären**, ob noch ein eigenes End-User-Handbuch erstellt werden soll oder ob die On-Screen-Hilfe ausreicht.

Stand: 2026-05-23 · Audit: Roland Neissl · Quelle: GitLab `dess/dess` Commit-Historie + Vault `wiki/concepts/integration-dess-openproject.md` + Vault `wiki/incidents/2026/2026-05-21-dess-sso-broken-after-kc26.md`

DESS · DB komplett zurücksetzen

DESS · DB komplett zurücksetzen

Achtung: Diese Anleitung löscht **alle Bewegungsdaten** in einer DESS-Instanz. Nur für Test-/Staging-Umgebungen oder bei einem expliziten Neuaufsatz nach Abstimmung. **Vorher Voll-Backup** der MariaDB-DB (siehe Plesk-Backup oder `mariadb-dump`).

Tabellen leeren

In MariaDB / MySQL `DROP TABLE` direkt ausführen — danach beginnen Auto-Increment-IDs wieder bei 1, Tabellen sind komplett leer.

Standard-Reset (Bewegungsdaten)

```
DROP TABLE tbl_notification;  
DROP TABLE tbl_on_call;  
DROP TABLE tbl_time_booking;  
DROP TABLE tbl_time_booking_change_log;  
DROP TABLE tbl_time_booking_day;  
DROP TABLE tbl_time_booking_hour;  
DROP TABLE tbl_time_booking_work;  
DROP TABLE tbl_log;  
DROP TABLE tbl_md1_entry;
```

Anschließend Anwendung **immer** neu starten (Plesk → Node.js → Restart App oder `touch tmp/restart.txt`). DESS legt die Tabellen beim Start mit TypeORM neu an.

Korrekturdaten beibehalten

Wenn nur Buchungen gelöscht werden sollen, MDL-/Abwesenheitskorrekturen aber bleiben sollen:

MDL behalten (nur Buchungs-MDL-Bezüge weg):

```
DELETE FROM tbl_md1_correction WHERE mdl_correction_booking_id <> 0;
```

Abwesenheitskorrekturen behalten (Urlaub, ZA):

```
DELETE FROM tbl_time_absence WHERE absence_booking_id <> 0;
```

Komplett-Reset inkl. MDL + Abwesenheits-Korrekturen

```
DROP TABLE tbl_md1_correction;  
DROP TABLE tbl_time_absence;
```

Pflicht-Schritt: Anwendung neu starten

```
touch /var/www/vhosts/agrarforschung.at/dess.agrarforschung.at/tmp/restart.txt
```

Oder per Plesk-UI: **Node.js → Restart App**. Ohne Neustart sind die fehlenden Tabellen für Node-Sessions noch im Cache und es kommt zu Fehlern.

Verifikation

Nach dem Neustart:

```
SELECT COUNT(*) FROM tbl_time_booking;  
SELECT MAX(booking_id) FROM tbl_time_booking;
```

Beide sollten 0 bzw. NULL liefern. Erste neue Buchung bekommt `booking_id = 1`.

