

# cDWH Dokumentation

Abel Hodelin Hernandez

14. Juli 2021

## Acronyms

<b>cDWH</b>	clinic Data Warehouse.....	2
<b>DB</b>	Datenbank .....	2
<b>TDE</b>	Transparent Data Encryption.....	2
<b>ETL</b>	Extraction Tranformation Load.....	12
<b>KIS</b>	Krankenhausinformationssystem.....	7
<b>GTDS</b>	Gißener Tumordokumentationssystem.....	13
<b>KHEntgG</b>	Krankenhausentgeltgesetz .....	12

# Inhaltsverzeichnis

<b>1</b>	<b>Installation und Konfiguration der Instanz des clinic Data Warehouse (cDWH)</b>	<b>4</b>
1.1	Transparent Data Encryption (TDE)-Installation . . . . .	4
1.2	TDE-Konfiguration . . . . .	5
1.2.1	TDE-Instanz-Dateien . . . . .	5
1.2.2	Konfigurationsdateien . . . . .	6
<b>2</b>	<b>Struktur des cDWH</b>	<b>7</b>
2.1	Beschreibung . . . . .	7
2.2	Liste der vorhandene Schemata . . . . .	7
<b>3</b>	<b>Nutzung</b>	<b>8</b>
3.1	Start die Datenbank (DB) . . . . .	8
3.2	Arbeiten mit cDWH via ssh/psql . . . . .	9
<b>4</b>	<b>Backup</b>	<b>10</b>
4.1	Konzept . . . . .	10
4.2	Technische Aspekte . . . . .	10
<b>5</b>	<b>Schemata</b>	<b>11</b>
5.1	p21 . . . . .	11
5.1.1	Tabellen . . . . .	11
5.1.2	Views . . . . .	12
5.2	kis . . . . .	12
5.2.1	Tabellen . . . . .	12
5.3	copra . . . . .	13
5.3.1	Tabellen . . . . .	13
5.4	gtds . . . . .	13
5.4.1	Tabellen . . . . .	13
5.5	centrallab . . . . .	14
5.5.1	Tabellen . . . . .	14

5.6	imagic . . . . .	14
5.6.1	Tabellen . . . . .	14
5.7	metadata_repository . . . . .	14
<b>6</b>	<b>Benutzer</b>	<b>15</b>

# Kapitel 1

## Installation und Konfiguration der Instanz des cDWH

Die PostgreSQL-Instanz wurde mit Hilfe von PostgreSQL TDE verschlüsselt. Das bedeutet, dass alle Dateien aus der PostgreSQL-Instanz sind auf dem Festplatte verschlüsselt gespeichert und bei lesen entschlüsselt.

### 1.1 TDE-Installion

Die Installation von PostgreSQL TDE Version postgresql-12.3-TDE\_1.0 verfolgte die Installation Guide des Software (<https://www.cybertec-postgresql.com/de/transparent-data-encryption-installation-guide/>) Davor waren die notwendige Bibliotheken und Software installiert:

- zlib1g-dev
- libssl-dev
- libldb-dev
- libldap2-dev
- libperl-dev
- python-dev
- libreadline-dev
- libxml2-dev
- libxslt1-dev

- bison
- flex
- uuid-dev
- make
- gcc
- libsystemd-dev
- libxml2-utils
- xsltproc

Install-Kommando:

```
sudo ./configure --prefix=/usr/local/pg12tde --with-openssl --with-perl --with-python --with-ldap --with-libxml --with-uuid=e2fs --with-systemd
```

Start-Kommando:

```
/usr/local/pg12tde/bin/initdb -D /media/db/cdw_database/clinic_instance
```

## 1.2 TDE-Konfiguration

### 1.2.1 TDE-Instanz-Dateien

Auf dem Betriebssystem wurde der Benutzer `clinicuser` angelegt, der die Administration der DB-Instanz und besitzt keine administrative Rechte auf dem Betriebssystem. Die Dateien der TDE-Instanz befinden sich in dem Server unter `/media/db/cdw_database`.

- `clinic_instance` – Ordner der Instanz der cDWH mit DB- und Konfigurationsdateien.
- `sh_scripts` – Shell-Skripts für den Schlüssel-Manager.
- `dbBack` – Täglicher Backup der ganzen Instanz.

### 1.2.2 Konfigurationsdateien

Die Datei postgresql.conf wurde wie folgt modifiziert:

- port = 5433 #Proxy der Instanz
- listen\_addresses = '\*' # Maschinen auf denen die Instanz abrufbar ist
- password\_encryption = scram-sha-256 # Kennwort-Verschlüsselung Protokoll
- encryption\_key\_command = '/media/db/cdw\_database/sh\_scripts/clinic\_instance\_key.sh'

In der Datei pg\_hba.conf wurden die Benutzer der Instanz definiert.

- local all all scram-sha-256 # lokale Verbindungen
- host all all 0.0.0.0/0 scram-sha-256 # externe Verbindungen

# Kapitel 2

## Struktur des cDWH

### 2.1 Beschreibung

Das cDWH besitzt zwei strukturell gleiche DBs, **staging** für die Produktion und **staging\_test** zum testen. Die DBs sind in verschiedenen Schemata geteilt, jede davon entspricht eine Quelle oder Aufgaben. Die Information der Schemata liegt in Kapitel 5.

### 2.2 Liste der vorhandene Schemata

Schema	Information
centrallab	Information aus dem Zentral Labor
copra	Information aus COPRA
gtDs	Information aus GTDS
icd_metadatainfo	ICD10-GM
kis	Information aus dem Krankenhausinformationssystem (KIS)
metadata_repository	Metadata
ops_metadatainfo	OPS
p21	Information von §21
aktin	Information des AKTIN-Projekts
diz_intern	Administrative Information



# Kapitel 3

## Nutzung

### 3.1 Start die DB

Die Instanz startet automatisch nach jedem Reboot des Server. Wenn die Instanz auf diese Weise nicht startet, sollte man folgendes machen / überprüfen:

- `ssh IP des Server -l cdw` # Login auf dem Server via ssh mit einem Benutzer mit sudo Rechten, unter Windows auch mit den Tools putty oder MobaXterm
- Die Partition der Instanz sollte automatisch gemountet werden, da es in fstab konfiguriert ist. Falls die Partition nicht gemountet ist, sollte man die folgende Schritte durchführen:
  - `cdw$ lsblk` # Überprüfen ob die Partition `/dev/sdb1` gemountet ist.
  - `cdw$ sudo mount /dev/sdb1 /media/db` # Falls die Partition `/dev/sdb1` nicht gemountet ist
- Die PostgreSQL-Instanz startet automatisch nach 100 Sekunden nach jedem Neu Start des Servers, da das Script zum Starten via cron-daemon abgerufen wird:
  - Befehl in crontab:  
`@reboot sleep 100 && /media/db/cdw_database/startDB.sh`

Falls die Instanz nicht automatisch startet sollte man diese Befehlen verfolgen.

- `cdw$ sudo su - clinicuser` # Benutzer ändern

- clinicuser\$ `cd /media/db/database` # Gehe zum Ordner der Instanz
- clinicuser\$ `/usr/local/pg12tde/bin/pg_ctl -D clinic_instance restart` # (Re)Start die Instanz

## 3.2 Arbeiten mit cDWH via ssh/psql

- `ssh Server_IP -l tooluser` # Login auf dem Server
- `tooluser$ /usr/local/pg12tde/bin/psql -p 5433 database_name -U user_name` # Verbindung mit einer Datenbank der Instanz
- `database_name#\c another_database_name` – Verbindung mit anderer DB

# Kapitel 4

## Backup

### 4.1 Konzept

Ein Dump der kompletten cDWH-Instanz wird täglich um 01:00 gemacht. Das sind zwei Prozeduren, erst verläuft `dumpall` der cDWH-Instanz und direkt danach werden die Backup-Dateien in einer ZIP-Datei verschlüsselt komprimiert. Diese Datei wird auf dem Server und auf einer extra-VM gespeichert.

### 4.2 Technische Aspekte

Ein Shell-Script garantiert die Speicherung und Verschlüsselung der cDWH-Instanz sowie die lokale und ferne Speicherung. Dieses Skript wird jeden Tag um 01:00 via cron-daemon abgerufen.

- Shell-Script: `backDB.sh`
- Befehl in crontab: `0 1 * * * /media/db/cdw_database/backDB.sh`
- Backup-Ordner: `/media/db/cdw_database/dbBack`
- Backup-Name-Format: `staging.YYYY-MM-DD.all.zip`

# Kapitel 5

## Schemata

Die Schemata speichern die "rohe" **pseudonymisierte** Information der ursprünglichen Systems oder die Metadaten. Diese Daten werden in Views analysiert oder weiter verarbeitet für andere Anwendungen oder Projekten. Wichtige Hinweis ist, dass die Daten in dem Data Warehouse bleiben unverändert.

### 5.1 p21

Dieses Schema speichert die jährliche Information der §21, die von Medizincontrolling in CSV-Dateien generiert wird.

Der jährliche Rhythmus ist zu groß, als dass die Daten bspw. zur Rekrutierung von Patienten für Studien aber auch zu Forschung genutzt werden können. Auf diesem Grund wird diese Information in der Zukunft nicht mehr aus CSV-Dateien genommen sondern direkt aus dem KIS.

#### 5.1.1 Tabellen

Tabelle 5.1: Tabellen im Schema p21

Tabelle	Beschreibung
p21_encounter	Information der Datei FALL.csv: Fälle
p21_department	Inhalt der Datei FAB.csv: Fachabteilung
p21_operation	Information der Datei OPS.csv: Operationen
p21_diagnosis	Basiert auf der Datei ICD.csv: Diagnosen (ICDGM-10)

### 5.1.2 Views

In diesem Schema befinden sich auch die Views für Extraction Transformation Load (ETL)-Prozessen die, solche Information aus §21 benötigen. Der Inhalt dieser Views entspricht die Formatierung der Krankenhausentgeltgesetz (KHEntgG) §21 Übermittlung und Nutzung der Daten.

Tabelle 5.2: Views im Schema p21

View	Beschreibung
<b>fall</b>	Falldaten
<b>fab</b>	Fachabteilungsangaben
<b>icd</b>	Diagnosenangaben
<b>ops</b>	Prozedurenangaben

## 5.2 kis

Hier werden die tagesaktuellen extrahierten Daten zu Patienten, Fällen, Bewegungen, Diagnosen und Prozeduren direkt aus dem Quellsystem KIS gespeichert. Mit Hilfe diesem Schema lassen sich viele der Abbildungen für weitere Projekte realisieren.

### 5.2.1 Tabellen

In diesem Schema behalten die Tabellen denselben Namen wie in KIS.

Tabelle 5.3: Tabellen im Schema kis

View	Beschreibung
<b>nbew</b>	Bewegung
<b>ndia</b>	Diagnosen
<b>nfal</b>	Fälle
<b>nicp</b>	Prozeduren
<b>npat</b>	Patienten
<b>norg</b>	Organisationseinheiten

## 5.3 copra

Hier wird die tagesaktuelle Information aus dem COPRA-System gespeichert. Dieses Schema beinhaltet Befunde, ärztliche Anweisungen und Überblick über Behandlungsschritte.

### 5.3.1 Tabellen

In diesem Schema behalten die Tabellen denselben Namen wie im COPRA-System.

Tabelle 5.4: Tabellen im Schema copra

Tabelle	Beschreibung
co6_data_decimal_6_3	Metadaten der numerischen Messungen
co6_data_object	Metadaten der Messungen von Typ Objekt
co6_medic_data_patient	Demografische Information der Patienten
co6_medic_pressure	Daten der Herz-Messungen

## 5.4 gtds

Dieses Schema speichert die Daten der mainzenen Instanz des Gißener Tumordokumentationssystem (GTDS) und somit die Erfassung und Verarbeitung der Daten der revidierten Basisdokumentation klinischen Krebsregistern.

### 5.4.1 Tabellen

Dieses Schema hat momentan nur eine Tabelle. Die ist auf eine View auf eine Auswertung auf die Daten des GTDS basiert.

Tabelle 5.5: Tabellen im Schema gtds

Tabelle	Beschreibung
auswertung_diz	Auswertung auf Daten auf GTDS

## 5.5 centrallab

Hier werden die Daten aus dem Zentrallabor (Institut für Klinische Chemie und Laboratoriumsmedizin) gespeichert.

### 5.5.1 Tabellen

Die Tabellen speichern die Messungen sowie Mapping zu LOINC-Code.

Tabelle 5.6: Tabellen im Schema centrallabor

Tabelle	Beschreibung
<code>observation</code>	Laborwerte der Patienten
<code>observationreport</code>	Verlinkung der Laborwerten mit Fälle und Patienten
<code>loinc_mapping_central_lab</code>	Mapping der LOINC-Code zu der Messungen und/Geräte

## 5.6 imagic

Hier wird die Information aus dem IMAGIC-System gespeichert. Dieses Schema beinhaltet Information aus der Hautklinik. Davon Metadaten der Bilder sowie Befunde anhand der Bilder.

### 5.6.1 Tabellen

In diesem Schema behalten die Tabellen denselben Namen wie im IMAGIC-System.

Tabelle 5.7: Tabellen im Schema imagic

Tabelle	Beschreibung
<code>image</code>	Metadaten der Bilder
<code>patient</code>	Patienten Informationen
<code>study</code>	Information der Studien an der Hautklinik
<code>visit</code>	Besuch/Fall-Information

## 5.7 metadata\_repository

# Kapitel 6

## Benutzer

Das cDWH hat auch verschiedene Benutzer mit unterschiedlichen Aufgaben.

Tabelle 6.1: Benutzer im cDWH

Benutzer	Schema	Berechtigungen	Anwendung
centrallabcdwuser	centrallab	select, insert, update, delete, truncate	-
clinicuser	all	administrator	-
grafana	all	select	ja
kisvendor	kis	select, insert, update, delete, truncate	-
p21user	p21	select, insert, update, delete, truncate	-
gtdscdwuser	gtdscdwuser	select, insert, update, delete, truncate	-
onlyreader	all in staging_test	select	-
copracdwuser	copra	select, insert, update, delete, truncate	-
imagiccdwuser	imagic	select, insert, update, delete, truncate	-