UNIVERSITÄTSmedizin.
MAINZ

Dokumentation

clinical staging Data Warehouse (csDWH)

Abel Hodelin Hernandez

überprüft von
Sami George Habib

23. September 2021

Inhaltsverzeichnis

Tabellenverzeichnis

Acronyms

DIZ Datenintegrationszentrum . . . 5
csDWH clinical staging Data Warehouse . . . 1
LTS Long-term support . . . 6
DB Datenbank . . . 5
TDE Transparent Data Encryption . . . 6
ZIP Zipper . . . 7
MD5 Message-Digest Algorithm 5 . . . 7
ICD-10-GM International Statistical Classification of Diseases and Related Health Problems, 10. Revision, German Modification . . . 9
ETL Extraction Tranformation Load . . . 14
KIS Krankenhausinformationssystem . . . 9
GTDS Gißener Tumordokumentationssystem . . . 9
KHEntgG Krankenhausentgeltgesetz . . . 14

Kapitel 1

Einführung

Im Datenintegrationszentrum (DIZ) werden Daten aus verschiedenen Fachabteilungen und Systemen zusammengeführt. Ein zentrales Puzzleteil für die Zwischenspeicherung der Information dieser Systemen ist das clinical staging Data Warehouse (csDWH). In dieser Datenbank (DB) werden alle relevanten klinischen Systeme abgebildet. Diese Daten werden im Rahmen des Datenschutz sowie der Datenqualität aufbereitet und anschließend an weitere Komponenten des DIZ übertragen.

Kapitel 2

Installation und Konfiguration der Instanz des csDWH

Das csDWH, welches die Forschungsdaten beinhaltet, befindet sich in einem Ubuntu Server mit der Version Ubuntu 18.04 Long-term support (LTS). Diese DB wurde in PostgreSQL mit Hilfe von PostgreSQL Transparent Data Encryption (TDE) implementiert und verschlüsselt. Somit sind alle Datensätze der Datenbank verschlüsselt auf der Festplatte gespeichert und werden erst bei Zugriff entschlüsselt.

2.1 TDE-Installation

Die Installation von PostgreSQL TDE Version postgresql-12.3_TDE_1.0 folgte dem Installation Guide Software unter dem Link (`https://www.cybertec-postgresql.com/de/transparent-data-encryption-installation-guide/`). Davor wurden die notwendigen Pakete und Abhängigkeiten auf dem Ubuntu-Server via apt installiert:

zlib1g-dev	libssl-dev
libldb-dev	libldap2-dev
libperl-dev	python-dev
libreadline-dev	libxml2-dev
libxslt1-dev	bison
flex	uuid-dev
make	make
gcc	libsystemd-dev
libxml2-utils	xsltproc

Das Install-Kommando lautet:
sudo ./configure –prefix=/usr/local/pg12tde –with-openssl –with-perl –with-python –with-ldap –with-libxml –with-uuid=e2fs –with-systemd

Das Start-Kommando für die Instanz lautet:
/usr/local/pg12tde/bin/initdb -D /media/db/cdw_database/clinic_instance

2.2 TDE-Konfiguration

2.2.1 TDE-Instanz-Dateien

Auf dem Betriebssystem wurde der Benutzer clinicuser angelegt, dieser ist für die Administration der DB-Instanz vorgesehen und besitzt keine administrative rechte auf dem Betriebssystem.

Die Dateien der TDE-Instanz befinden sich auf dem Server unter /media/db/cdw_database.

- clinic_instance – Instanz der csDWH mit DB- und Konfigurationsdateien.
- sh_scripts – Shell-Skripts.
  - clinic_instance_key.sh – Skript fürs Schlüssel-Manager. Der Schlüssel der Instanz ist ein Message-Digest Algorithm 5 (MD5)-Hash.
- dbBack – Täglicher Backup der ganzen Instanz. Hier werden die fünf letzten Backups der DB in verschlüsselten Zipper (ZIP)-Dateien aufbewahrt. Die Namenskonvention für die Backup Dateien ist staging_YYYY-MM-TT.all.zip.

2.2.2 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' # Datei mit dem Schlüssel der Instanz

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 3

Struktur des csDWH

3.1 Beschreibung

Das csDWH besitzt zwei strukturell gleiche DB, `staging` für die Produktion und `staging_test` zum testen. Die DB sind in verschiedenen Schemata geteilt, jede davon entspricht eine Quelle oder Aufgaben. Die Information der Schemata liegt in Kapitel 6.

3.2 Liste der vorhandenen Schemata

Tabelle 3.1: Schemata im csDWH

Schema	Information
centrallab	Information aus dem Zentral Labor
copra	Information aus COPRA-System (PDMS)
gtds	Information aus dem Gißener Tumordokumentationssystem (GTDS)
icd_metadatainfo	International Statistical Classification of Diseases and Related Health Problems, 10. Revision, German Modification (ICD-10-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 4

Nutzung

4.1 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

4.2 Arbeiten mit csDWH 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 5

Backup

5.1 Konzept

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

5.2 Technische Aspekte

Ein Shell-Script garantiert die Speicherung und Verschlüsselung der csDWH-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 6

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.

6.1 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.

6.1.1 Tabellen

Tabelle 6.1: 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 (ICD-10-GM)

6.1.2 Views

In diesem Schema befinden sich auch die Views für Extraction Tranformation 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 6.2: Views im Schema p21

View	Beschreibung
`fall`	Falldaten
`fab`	Fachabteilungsangaben
`icd`	Diagnosenangaben
`ops`	Prozedurenangaben

6.2 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.

6.2.1 Tabellen

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

Tabelle 6.3: Tabellen im Schema kis

View	Beschreibung
`nbew`	Bewegung
`ndia`	Diagnosen
`nfal`	Fälle
`nicp`	Prozeduren
`npat`	Patienten
`norg`	Organisationseinheiten

6.3 copra

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

6.3.1 Tabellen

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

Tabelle 6.4: Tabellen im Schema copra

Tabelle	Beschreibung
co6_data_decimal_6_3	Metadaten der nummerischen 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

6.4 gtds

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

6.4.1 Tabellen

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

Tabelle 6.5: Tabellen im Schema gtds

Tabelle	Beschreibung
auswertung_diz	Auswertung auf Daten auf GTDS

6.5 centrallab

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

6.5.1 Tabellen

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

Tabelle 6.6: Tabellen im Schema centrallabor

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

6.6 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.

6.6.1 Tabellen

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

Tabelle 6.7: Tabellen im Schema imagic

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

6.7 metadata_repository

Kapitel 7

Benutzer

Das csDWH hat auch verschiedene Benutzer mit unterschiedlichen Aufgaben.

Tabelle 7.1: Benutzer im csDWH

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	-