

Sisukord

- Sissejuhatus
- 1 - Seotud dokumendid ja täiendav materjal (Confluence)
- 2 - STEEL TERK andmeaida komponentide vaade
 - CI/CD
 - Platform ja käivitused
 - Andmeallikad
 - VERTICA STEEL andmeait
 - Andmehaldus ja jälgimine
- 3 - Andmelaadimiste töövoog
 - Tegevussammude sisukirjeldus
 - ODS DDL rakendamine
 - SAP -> ODS andmelaadimised
 - dbt transformatsioonid
- 4 - Andmevoog ja kihiline arhitektuur STEEL TERK andmeaidas
 - Andmevoog tervikuna
 - ODS – Operatiivandmete koopia
 - EDW – Ärimudel Verticas
 - DWH – Aruandluse kiht
- 5 - Keskkondade loogika STEEL TERK andmeplatvormil
 - Git–CI/CD loogika ja konteinerite ehitamine
 - Arenduse töövoog
 - Test-keskkonna CI/CD
 - Live-keskkonna CI/CD
 - Kubernetes ja keskkondade eraldatus
 - Vertica keskkonnad ja skeemid
 - Arenduse skeemid Verticas
 - Test-keskkonna skeemid Verticas
 - Live-keskkond Verticas
 - SAP HANA allikad test- ja live-keskkondades

Sissejuhatus

STEEL TERK andmeplatvorm on Tervisekassa (TERK) andmelao tehniline tuumik, mis koondab SAP HANA allikatest pärinevad operatiivandmed ühtsesse, standardiseeritud ja versiooneeritud

andmehoidlasse. Platvormi eesmärk on pakkuda usaldusväärset, korduvkasutatavat ja hästi hallatud andmebaasi, mis toetab nii aruandlust, analüüsi kui ka edasist äri loogika arendamist.

Käesolev dokument kirjeldab STEEL TERK andmelao arhitektuuri, tehnilisi komponente, keskkondade loogikat, laadimiste ja transformatsioonide protsesse ning CI/CD toimimist. Fookus on tehnilistel mehhanismidel – kuidas andmed liiguvad SAP süsteemidest ODS kihti, edasi EDW mudelitesse ning lõpuks DWH aruandluskihile; kuidas Kestra, Kubernetes, dbt, Apache Hop ja Liquibase koos töötavad; ning kuidas arendus-, test- ja live-keskkonnad on üles ehitatud.

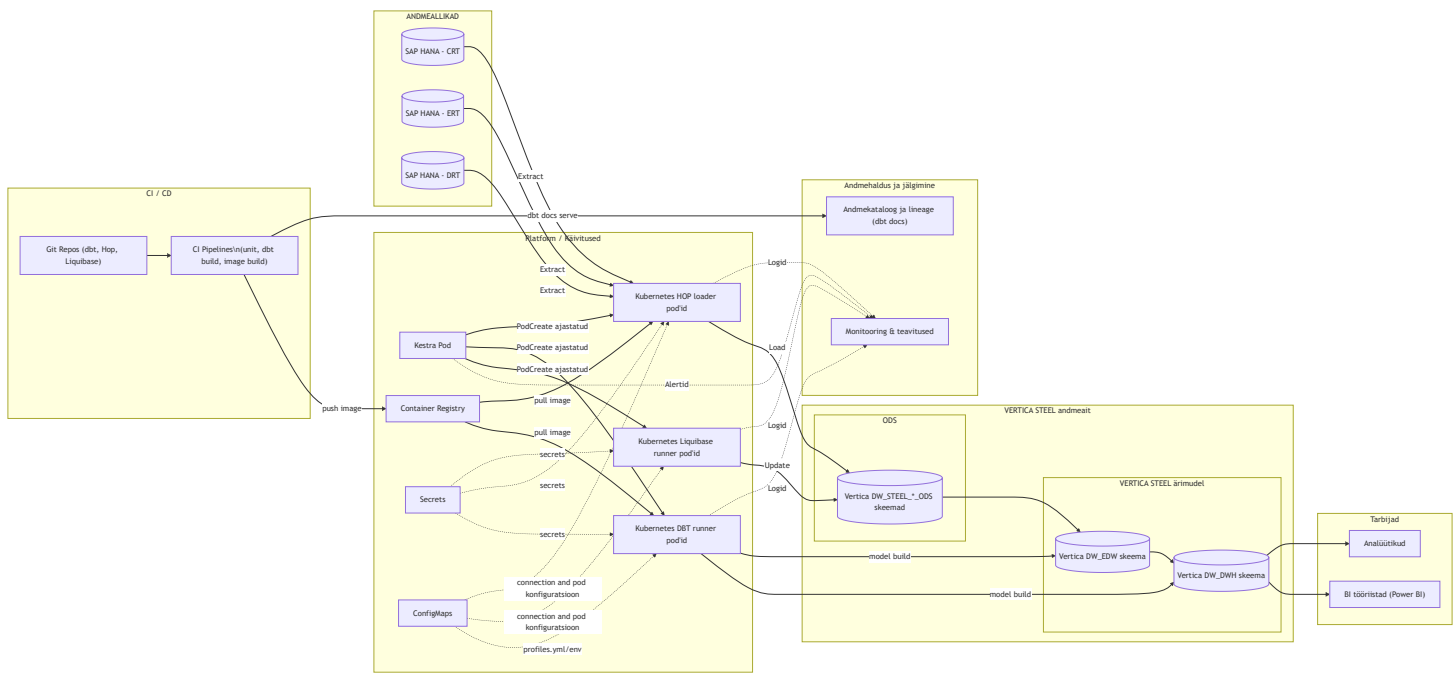
Dokument ei dubleeri eraldi Confluence'i dokumentatsiooni, mis katab detailsemalt andmekonversioone, õiguste haldust, testimist ja ärimõisteid. Need on viidatud eraldi peatükis ning integreeritud temaatiliste osade alla.

1 - Seotud dokumendid ja täiendav materjal (Confluence)

- Andmekonversioonid Allikas -> ODS
- Andmelao õiguste maatriks
- Andmelao testimise plaan
- Andmelao ärisõnastik
- dbt kasutusloogika
- Projekti nõuded

2 - STEEL TERK andmeaida komponentide vaade

- Eesmärk: Kirjeldada STEEL TERK andmeplatformi ehitusel kasutatavaid komponente, nende eesmärged ning kasutust
- Koostaja: Andres Kukk (<mailto:andres.kukk@wizon.ee>)
- Kuupäev: 2025-11-24



CI/CD

Projektis kasutatavad git repod

- Liquibase - kasutusel DDL haldamiseks ODS kihi tabelite jaoks. Igale alliksüsteemile vastavad andmed hoitakse ODS kihi eraldi skeemas. Vastavalt allikatele/ODS-skeemadele on jagatud ka DDL kood kolme repo vahel:
 - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dw_steel_crm_ods_ddl
 - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dw_steel_herp_ods_ddl
 - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dw_steel_derp_ods_ddl
 - CI töövoos käigus ehitatakse liquibase töö-image'd mis sisaldavad DDL koodi ning Liquibase changelog'i. Töö-image laaditakse TEHIKU image-registrisse
registry.tehik.ee/dwh/domains/tis/steel
- Apache Hop - kasutusel andmete pärimiseks alliksüsteemidest ning nende kirjutamiseks STEEL andmeaita (extract-load). Igale alliksüsteemile vastavad andmed hoitakse ODS kihi eraldi skeemas. Vastavalt allikatele/ODS-skeemadele on jagatud ka DDL kood kolme repo vahel:
 - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dw_steel_crm_ods
 - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dw_steel_herp_ods
 - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dw_steel_derp_ods
 - CI töövoos käigus ehitatakse hop-loader image'd mis sisaldavad kasutatavat Apache Hop koodi (asukohas /files ja /files/hop-loader-helpers) ning laadimisel kasutatavat SQL koodi (extract-päringud, STG-ODS laadimised; alamkaustas /files/ODS/).
- Dbt - kasutusel on kolm dbt repot.
 - dbt central - <https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dbt-central>

- repo on kasutusel selleks, et liidestada omavahel järgnevad child-repod (dbt deps). Selle repo sees ei ole hetkel mudelite koodi. Siin võib hoida keskset haldust nõudvat osa dbt koodist - näiteks testid, macrod mida peab saama kasutada kõikides child-repodes ning mida ei ole mõistlik igas child-repos eraldi hallata. CI töövoos käigus ehitatakse selle repo baasilt dbt-runner image. Image sisaldab ainult dbt-central repos sisalduvat koodi ning dbt konfiguratsiooni template'i. Child repod mapitakse konteineri külge gitist runtime'ile.
- EDW_STEEL - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/edw_steel
 - repo sisaldab dbt koodi EDW_STEEL skeema tarbeks. Kood asub alamkataloogis /files/terk-steel
 - /snapshots - kood andmeajaloo säilitamiseks andmeidas dbt snapshot funktsionaalsuse abil. See on transformatsiooni-ahela esimene samm kõikide ODS tabelite tarbeks.
 - /models/base - kood aruandluseks vajalike andmete selekteerimise sammu tarbeks.
 - /models/core - kood transformatsiooni-ahela vahesammude tarbeks.
 - /models/final - kood transformatsiooniahela lõpp sammude tarbeks (veergude ja tabelinimede tõlkimine ärikasutuse keelde)
 - /seeds - csv failid aruandluseks vajaliku lisisisendi laadimiseks andmeaita (EI OLE KASUTUSEL)
- DWH_STEEL - https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/dwh_steel
 - repo sisaldab dbt koodi DWH_STEEL skeema tarbeks. Kood asub alamkataloogis /files/terk-steel
- Kestra
 - <https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/kestra> - repo sisaldab DDLi, HOP laadimiste ning dbt transformatsioonide käivituseks mõeldud Kestra flow'de koodi. Samuti laadimiste järjestuse loogikat, käivituste ajastamist ning teavituste haldamise kärke Kestra tarbeks.
- Kubernetes
 - <https://gitlab.sotsiaalministeerium.ee/dwh/domains/tis/steel/kubernetes> - repo sisaldab secrets.yaml ning configmaps.yaml failide template'i. Need on kasutusel Liquibase, Hop ning dbt podide seadistuste ning saladuste haldamisel.

Platform ja käivitused

STEEL andmeplatvorm töötab Kubernetesel, kus iga tööülesanne on eraldi konteineripõhine käivitusüksus (ephemeral pod). Kestra vastutab kõikide laadimiste, transformatsioonide ja DDL muudatuste ajastamise ja orkestreerimise eest. Diagrammil nähtavad PodCreate nooled tähistavad seda, et Kestra loob HOP-i, Liquibase'i ja dbt podid vastavalt laadimisgraafikule või käsitsi käivitustele.

HOP-loader podid tegelevad SAP HANA allikate (CRT, DRT, ERT) päringute tegemisega ja andmete laadimisega ODS-skeemidesse.

Liquibase podid haldavad ODS-kihi tabelite versioone ja muudatusi.

Dbt runner podid tegelevad ärimudeli loomisega – ODS → EDW → DWH transformatsioonid leiavad aset mudelite ehitamise (dbt build) käigus.

Kõik podid tõmbavad oma image'd image-registrist, mis tagab, et kasutusel on see versioon, mille CI pipeline on tootnud. Saladused tulevad Kubernetes secrets'itest ning ühendused ja käivitusseaded ConfigMap'idest. Logid suunatakse monitooringusüsteemi, kust saadakse ka teavitused tõrgete korral.

Andmeallikad

Diagrammil kujutatud SAP HANA allikad – CRT, DRT ja ERT – on STEEL platvormi esmased sisendtarnijad. Need süsteemid sisaldavad operatiiv-andmeid, mida laaditakse andmeaita muutumatul kujul (tabel allikas ning vastav allikas STEEL andmelao ODS skeemas peaks olema sama andmekoosseisuga). HOP-loader pod'id teevad extracte HANA tabelitest, kasutades ConfigMap'ides määratud ühendusi ja secretist saadavaid ühemduse andmeid.

Iga allikas laaditakse talle vastavasse ODS-skeemasse, mis hoiab allikapõhise eristuse selgelt paigas.

VERTICA STEEL andmeait

STEEL andmeait on üles ehitatud kolmekihilisena – ODS, EDW ja DWH. ODS hoiab allikatest tulnud andmed muutmata kujul. Liquibase pod'id hoiavad ODS-i tabelid nõuetekohasel kujul, HOP pod'id laadivad sinna andmed. EDW kiht on kasutusel äri loogika rakendamiseks ja DWH kiht andmete presenteerimiseks ning kasutamiseks.

Dbt runner pod'id ehitavad mudelid ODS-ist EDW-sse ja sealt edasi DWH-sse ("model build" joonisel). EDW-st saab alguse transformeeritud, tehniliselt korrastatud andmevaade. DWH on seevastu kasutajatele mõeldud tõlgitud ärikiht, millele toetuvad nii BI kui analüütikud.

Andmehaldus ja jälgimine

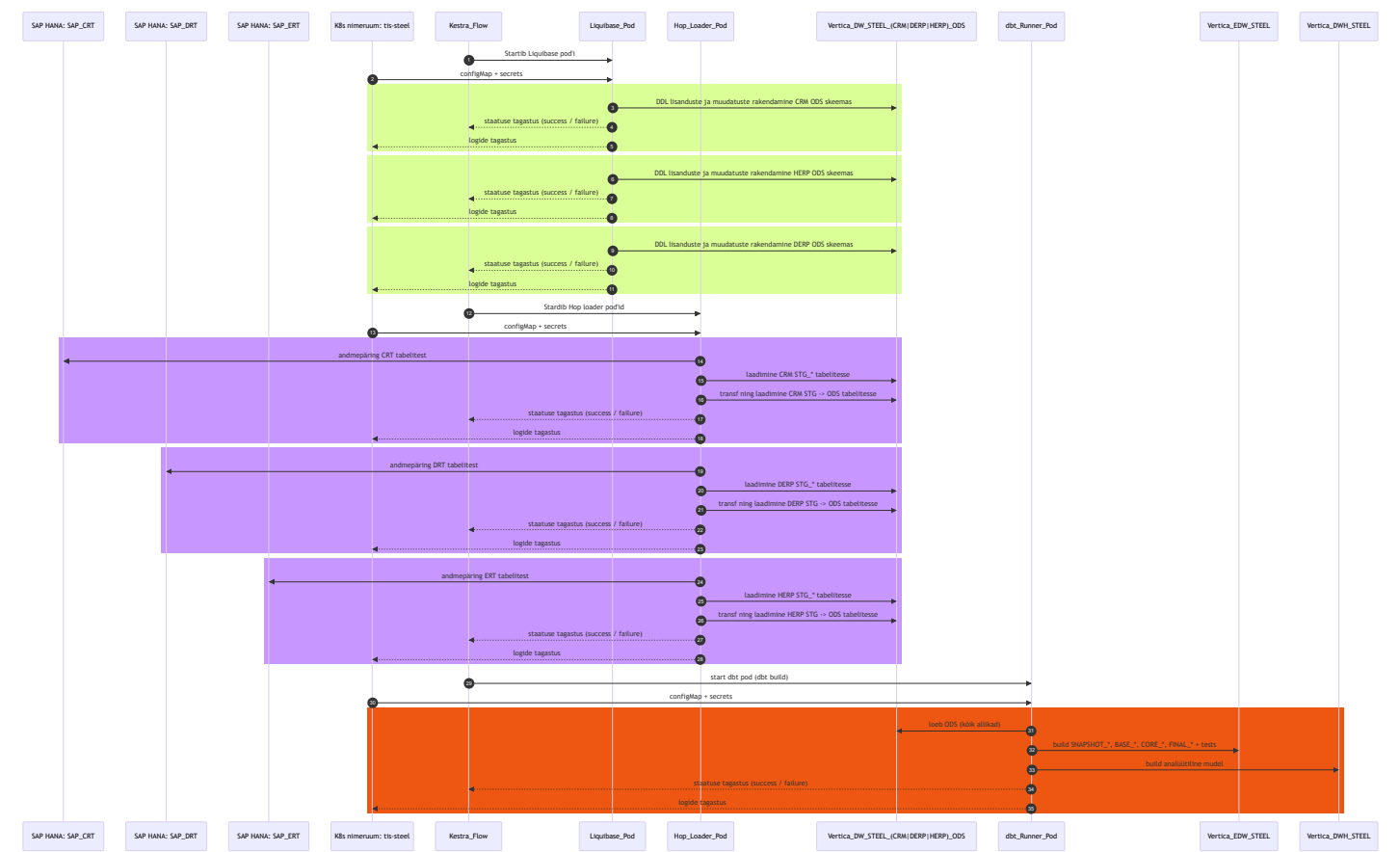
Diagrammil on eraldi välja toodud andmekataloog ja lineage (dbt docs) ning monitooring – need on platvormi läbivad tugiteenused.

Dbt-l põhinev dokumentatsioon tekib CI töövoos (dbt docs generate) ning teenindatakse CI kaudu ("dbt docs serve"). See tagab, et mudelite kirjeldus ja lineage on alati versioonikontrolli seisuga kooskõlas. Andmekataloog näitab, mis mudelid on STEEL andmeaidas ja kuidas need üksteisest sõltuvad (data lineage) samuti hoitakse seal ärilist kirjeldust objektide tähenduse kohta.

Monitooring kogub logisid HOP-, dbt- ja Liquibase-podidelt (diagrammil näha punktiirnooled). Kestra saab teavitusi laadimiste ja transformatsioonide seisu kohta ka andmeida halduse e-mailidele ja või webhookidele.

3 - Andmelaadimiste töövoog

- Eesmärk: Kirjeldada andmelaadimiste ning transformatsioonide käigus tehtavaid tegevussamme kasutatava tehnilise seadistuse kontekstis.
- Koostaja: Andres Kukk (<mailto:andres.kukk@wizon.ee>)
- Kuupäev: 2025-11-24



VÄRVID JOONISEL

Märge	Kirjeldus
	ODS DDL rakendamine
	SAP → ODS andmelaadimised
	dbt transformatsioonid

Tegevussammude sisukirjeldus:

ODS DDL rakendamine

1. Liquibase podid ehitatakse allikapõhistelt liquibase-imagetelt. Need ehitatakse vastava repo CI töövoos käigus ning sisaldavad ODS kihi ddl koodi. Iga allikapõhise ODS-skeema loomiseks/muutmiseks luuakse eraldi Liquibase-pod (ephemeral) mis pärast täitmist sulgetakse. Käivituste järjestust ning paralleelsust hallatakse Kestra töövoos (flow) tasemel. Koodi hoitakse järgnevatel repodes:
 - repo: /tis/steel/kestra
 - repo: /tis/steel/dw_steel_crm_ods_ddl
 - repo: /tis/steel/dw_steel_herp_ods_ddl
 - repo: /tis/steel/dw_steel_derp_ods_ddl
2. K8s configMaps ning secrets failid on kasutusel Liquibase-pod'ide seadistuste ning ligipääsude juhtimiseks. Vastavate failide struktuur asub repos:
 - repo: /tis/steel/kubernetes
 - configmap/secrets: dw-steel-crm-ods-ddl
 - configmap/secrets: dw-steel-herp-ods-ddl
 - configmap/secrets: dw-steel-derp-ods-ddl
3. Liquibase loob/ning muudab ODS skeemade objektid vastavalt ddl sisendile. Logib muudatused.
4. Liquibase käivituse staatust ning logi edastatakse Kestrasse (pikemaajalist logide säilitamist siin ei toimu)
5. Liquibase käivituse staatust ning logi kogutakse kesksesse logirakendusse otse podidelt.

SAP -> ODS andmelaadimised

1. Hop podid ehitatakse allikapõhistelt hop-imagetelt. Need ehitatakse vastava repo CI töövoos käigus ning sisaldavad allika tabeli/vaatepõhiseid päringuid ning stg->ods loogikat. Iga tabeli laadimiseks luuakse eraldi Hop-pod (ephemeral) mis pärast laadimise lõppemist sulgetakse. Laadimiste järjestust ning paralleelsust hallatakse Kestra töövoos (flow) tasemel. Koodi hoitakse järgnevatel repodes:
 - repo: /tis/steel/kestra
 - repo: /tis/steel/dw_steel_crm_ods
 - repo: /tis/steel/dw_steel_herp_ods
 - repo: /tis/steel/dw_steel_derp_ods

2. K8s configMaps ning secrets failid on kasutusel hop-pod'ide seadistuste ning ligipääsude juhtimiseks. Vastavate failide struktuur asub repos:

- repo: /tis/steel/kubernetes
- configmap: hop-loader-config-crm
- configmap: hop-loader-config-herp
- configmap: hop-loader-config-derp
- secrets: hop-loader-secrets-crm
- secrets: hop-loader-secrets-herp
- secrets: hop-loader-secrets-derp

3. Andmelaadimised allikatest STEEL andmeaida ODS skeemadesse (allikapõhised):

- laadimised toimuvad copy stream põhimõttel
- kõikide tabelite tarbeks on seadistatud ka inkrementaalse laadimise valmisolek. Regulaarsel laadimisel on see kasutusel ainult tabelite tarbeks millel on olemas primaarvõti ning ajatempel mille abil delta-laadimist juhtida. Inkrementaal-/täislaadimised on eristatud Kestra flow tasemel.

4. Laadimise staatust ning logi edastatakse Kestrasse (pikemaajalist logide säilitamist siin ei toimu)

5. Laadimiste staatus ning logi kogutakse kesksesse logirakendusse otse podidelt.

dbt transformatsioonid

1. Hop laadimiste lõppemisel käivitab kestra dbt pod'i. See ehitatakse /tis/steel/dbt-central repo CI töövoos.

- repo: /tis/steel/dbt-central - sisaldab ainult kesket haldamist vajavaid osi dbt loogikast (testid, macrod jmt).
- child-repo: /tis/steel/edw-steel - sisaldab dbt mudeleid EDW kihi kohta. Mapitakse gitist runtime'le.
- child-repo: /tis/steel/dwh-steel - sisaldab dbt mudeleid DWH kihi kohta. Mapitakse gitist runtime'le.

2. dbt käivitatakse `build` käsuga mis käivitab sõltuva jadana dbt testid, snapshotid, model'id.

3. dbt käivituse staatust ning logi edastatakse Kestrasse (pikemaajalist logide säilitamist siin ei toimu)

4. dbt käivituse staatus ning logi kogutakse kesksesse logirakendusse otse podidelt.

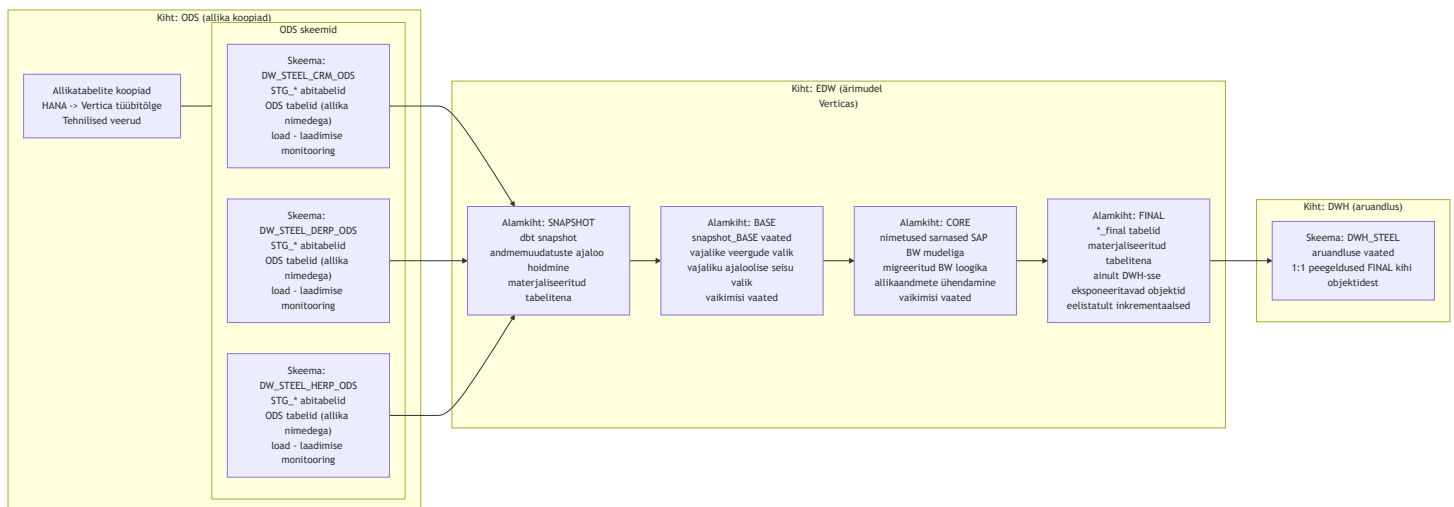
4 - Andmevoog ja kihiline arhitektuur STEEL TERK andmeaidas

- Eesmärk: Kirjeldada andmeaida kihilist arhitektuuri ning kirjeldada andmeaida kihtide sisu ning eesmäärke.
- Koostaja: Andres Kukk (<mailto:andres.kukk@wizon.ee>) ja
- Kuupäev: 2025-11-25

STEEL TERK andmeaida arhitektuur on üles ehitatud kolmekihilise mudelina. Kihid ja nende rollid:

- ODS (operatiivandmete koopia),
- EDW (ärimudel Verticas) ja
- DWH (aruandluse kiht).

Diagramm kirjeldab, kuidas andmed liiguvad allikasüsteemidest läbi erinevate transformatsioonisammude kuni lõppkasutajani.



Andmevoog tervikuna

Diagrammil kujutatud nooled kirjeldavad ühesuunalist andmeteed:

1. **ODS skeemid** → **EDW SNAPSHOT**
2. SNAPSHOT → BASE
3. BASE → CORE
4. CORE → FINAL
5. FINAL → **DWH**

See ülesehitus tagab:

- selge vastutusjaotuse (allika koopia, transformatsioon, eksponeerimine),
- auditeeritava andme-lineage'i,
- standardiseeritud andmemudeli, mis toetab usaldusväärset aruandlust ja analüüsi.

ODS – Operatiivandmete koopia

ODS kiht on tehniline puhver SAP HANA (tulevikus võib allikaid lisanduda) allikate ja ülejäänud andmeplatvormi vahel. Siin hoitakse allikatabelite võimalikult üks-ühele koopiad koos vajalike tehniliste rikastustega. Sel kihil toimub:

- **andmetüüpide ühtlustamine** HANA → Vertica vormingute vahel,
- **tehniliste veergude lisamine**,
- **tabelite laadimise monitooring** selleks tarbeks load tabelid kõikides ODS kihi skeemades.

ODS kiht koosneb alliksüsteemi põhiselt loodud skeemadest. Hetkeseisuga kuulub sinna kolm skeemat — DW_STEEL_CRM_ODS, DW_STEEL_DERP_ODS ja DW_STEEL_HERP_ODS. Igas skeemas on STG_* laadimise abitabelid ja ODS tabelid, mis kasutavad allikaga sama nimekuju. ODS on ühtlasi lähtepunkt EDW kihi SNAPSHOT alamkihile.

STG_ tabelid on kasutusel ainult laadimiste juhtimiseks. Eduka laadimise lõpus need tühjendatakse. Laadimise vea korral STG_ -> ODS sammus jääb STG_ tabelitesse ebaõnnestunud laadimise delta. Seda saab kasutada veaotsingul.

ODS kihti kuuluvad tabelid peaksid 1:1 vastama allikale. Vastavad peavad olema ridade arv, veergude arv kui väljade sisu. Ainsad muudatused võrreldes allikaga:

- veerunimekujude muutmine vastavaks Vertica baasi reeglitele - nt kaldkriipsud tabeli ning veerunimeses on asendatud alakriipsudega
- lisatud on laadimiste metaandmete veerud mis näitavad rea andmeaita jõudmise kuupäeva ning laadimisprotsessi id'd.
- tehtud on andmetüübiteisendused allikas vs sihtbaas.

EDW – Ärimudel Verticas

EDW kiht moodustab andmehoidla keskse ärioloogilise mudeli. Kihil on neli alamkihti, millest igaüks täidab kindla etapi andmetöötlus-ahelas:

- **SNAPSHOT** – säilitab ajaloolised andmemuudatused dbt snapshot mehhanismiga; tulemusena tekivad materjaliseeritud versiooni-tabelid (SCD) iga allikast laaditud tabeli kohta. Kuna ODS on allika täpne koopia siis tekitatakse muudatuste ajalugu SNAPSHOT-alamkihis. ETTEVAATUST: tegevused SNAPSHOT kihis võivad tekitada püsivat andmekadu!

- **BASE** – valib vajalikud veerud ja ajaloolise seisu ning loob alusteabe EDW mudelile; koosneb andmebaasi vaadetest. Sisulisi andmeteisendusi siin ei tehta.
- **CORE** – migreeritud SAP BW loogika ning vastavalt TERK analüütikute sisendile tehtud muudatused selles, ühendab allikad ja annab mudelile ärilise struktuuri; ärilooika rakendmine toimub siin kihis; vaikimisi implementeeritud andmebaasi-vaadetena.
- **FINAL** – lõplikud materjaliseeritud tabelid, mis on mõeldud DWH kihi eksponeerimiseks; eelistatult inkrementaalsed mudelid. Iga objekt mida peegeldatakse DWH kihti peaks EDW kihis omama FINAL-kihi tabelit. SAP BW tabelite ning veergude tõlkimine ärikeeelde toimub siin kihis (1:1 migreeritavate objektide korral)

Andmevoog liigub alamkihtide vahel sirgjooneliselt: **SNAPSHOT** → **BASE** → **CORE** → **FINAL**.

DWH – Aruandluse kiht

DWH kiht on kasutajatele eksponeeritav aruandluskiht. Skeema `DWH_STEEL` sisaldab:

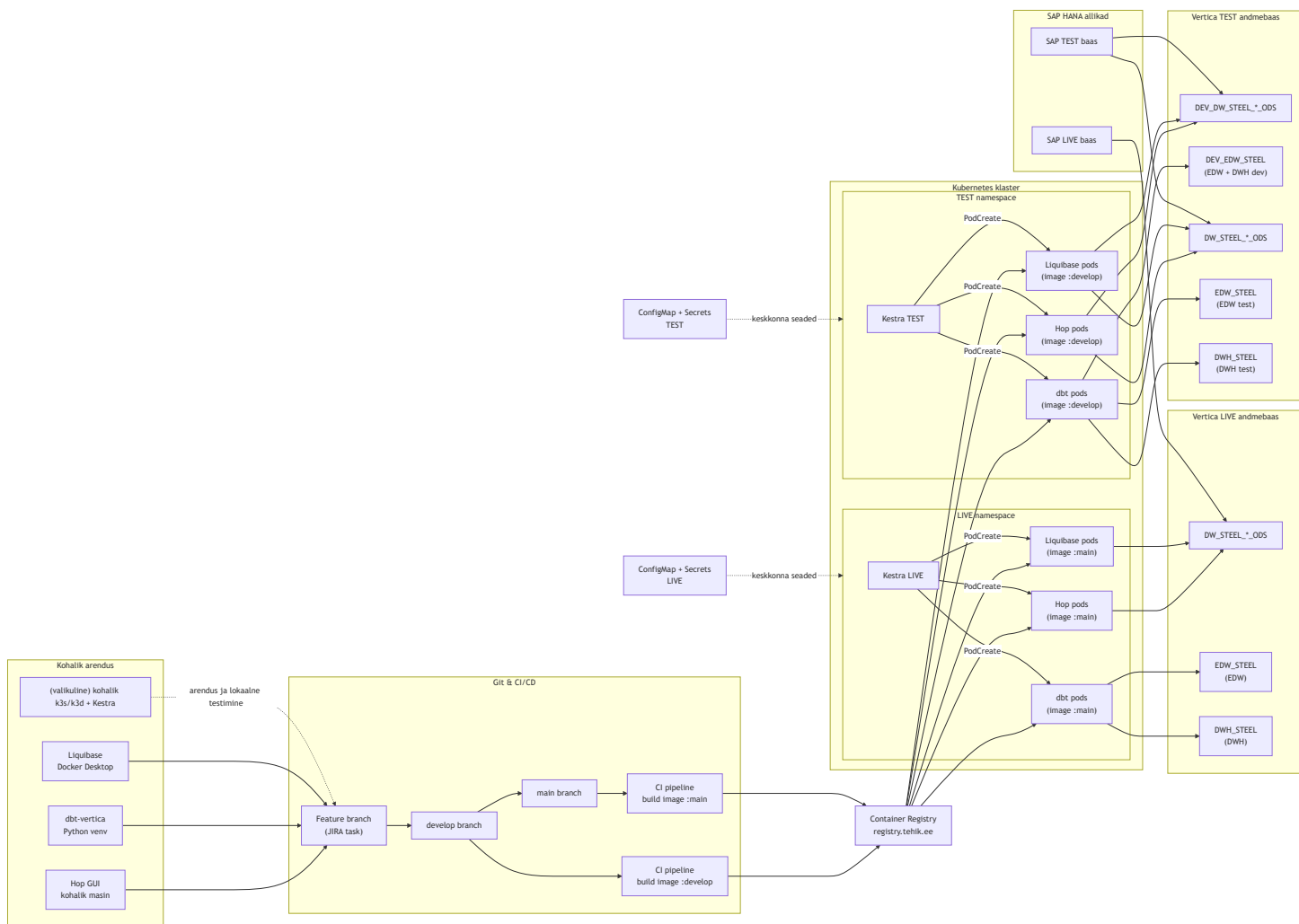
- **aruandluse vaateid - 1:1 peegeldusi EDW FINAL mudelitest,**
- vormi ja keelestandardit, mis sobib BI tööriistadele ja analüütikutele.

DWH kiht ei sisalda täiendavat ärilooikat – see on EDW FINAL-objektide esitluskiht.

5 - Keskkondade loogika STEEL TERK andmeplatvormil

STEEL andmeplatvorm kasutab kaheetapilist keskkonnamudelit: **TEST** ja **LIVE**. Arenduseks eraldi keskkonda ei ole – kogu arendus toimub lokaalselt arendajate töövahendites ning Git feature-branchide kaudu. Test- ja live-keskkonnad töötavad Kubernetesel ning on eristatud läbi konteineri-image, configmap'ide ja secrete'ite.

Alljärgnevalt on kirjeldatud, kuidas toimub versioonihaldus, konteinerite ehitamine, loomine ja käivitamine, ning kuidas on jagatud andmebaasi- ja SAP-allikakeskkonnad.



Git-CI/CD loogika ja konteinerite ehitamine

Arenduse töövoog

Arendamiseks ei ole eraldi serverikeskkonda. Koodi arendus toimub lokaalselt:

- **HOP** – arendus Hop GUI abil arendaja arvutis.
- **dbt** – arendus Python virtuaalkeskkonnas dbt-verticaga.
- **Liquibase** – DDL arendamine ja testimine Docker Desktopi abil.
- **Kestra** – vajadusel võimalik käivitada lokaalne k3s/k3d keskkond Kestra jooksutamiseks.

Arendaja töövoog:

1. Arendaja loob **feature-branchi** (nimega vastavalt JIRA taskile).
2. Arendused tehakse lokaalselt, testitakse lokaalselt.
3. Feature-branchist tehakse **merge request** → develop .

Test-keskkonna CI/CD

Kui merge request kinnitatakse ja kood jõuab `develop` harusse:

1. CI pipeline **ehitab tööimage** (HOP, dbt runner, Liquibase).
2. Image märgistatakse tagiga `:develop`.
3. Image pushitakse TEHIK container registry'sse.
4. Kestra TEST keskkond kasutab ephemeral podide loomiseks alati `:develop` tagiga image'e.

Test-keskkond on seega **arenduse ametlik kontrollpunkt**, kus testitakse:

- laadimised,
- dbt mudelite ehitamine,
- DDL muudatused,
- Kestra töövoogude korrektne toimimine,
- andmevoogude terviklikkus.

Live-keskkonna CI/CD

Kui test-keskkonnas on kõik kinnitatud ning `develop` → `main` merge on tehtud:

1. CI pipeline ehitab uue tööimage **main-branchi seisust**.
2. Image märgistatakse tagiga `:main`.
3. Image pushitakse registrisse.
4. Kestra LIVE keskkond kasutab ephemeral podide loomiseks `:main` tagiga image'e.

Live saab seega **ainult CI kaudu**, mitte kunagi käsitsi.

Kubernetes ja keskkondade eraldatus

TEST ja LIVE keskkonnad jooksevad eraldi Kubernetes namespace'ides.

Keskkondade eraldatus saavutatakse läbi:

- eraldi **ConfigMapide**,
- eraldi **Secretite**,
- eraldi **image tagide** (`:develop` ja `:main`),
- eraldi andmebaasiskeemide,
- eraldi SAP-allikate kasutuse.

Igal keskkonnal on:

- oma ühendusparameetrid,
- oma salajased võtmed (secretid),
- oma schedulerid,
- oma Kestra vood,
- oma pod'ide konfigureerimine (ressursid, ajastus, connection-id'd).

TEST ja LIVE ei jaga mitte midagi – isegi mitte ajastusi.

Vertica keskkonnad ja skeemid

Vertica-s on **kolm loogilist tegevusruumi**:

1. **Arendus** – toimub test-andmebaasi *arendusskeemides*.
2. **Test** – toimub test-andmebaasi *testskeemides*.
3. **Live** – eraldi prod-andmebaasis.

Arenduse skeemid Verticas

Arendus ei toimu eraldi serveris, vaid **test-baasi arendusskeemides**:

- ODS:
 - DEV_DW_STEEL_HERP_ODS
 - DEV_DW_STEEL_CRM_ODS
 - DEV_DW_STEEL_DERP_ODS
- EDW:
 - DEV_EDW_STEEL
- DWH:
 - DEV_EDW_STEEL (sama skeema sisaldab DWH objekti taset)

Siin saavad arendajad jooksutada test-laadimisi, teste ja dbt-mudeleid.

Test-keskkonna skeemid Verticas

Testimiseks kasutatakse *sama andmebaasi eraldi skeeme*:

- ODS:
 - DW_STEEL_HERP_ODS
 - DW_STEEL_CRM_ODS
 - DW_STEEL_DERP_ODS
- EDW:

- EDW_STEEL
- DWH:
 - EDW_STEEL (DWH osa samas skeemas)

Need skeemid töötavad koos Kubernetes TEST image'idega ja Kestra TEST ajastusega.

Live-keskkond Verticas

Live töötab **eraldi Vertica andmebaasis**, kus skeemide nimetused on testiga identsed:

- DW_STEEL_HERP_ODS
- DW_STEEL_CRM_ODS
- DW_STEEL_DERP_ODS
- EDW_STEEL

See tagab, et deploymentid on **skeema-tasandil 1:1 ülekantavad** test → live.

SAP HANA allikad test- ja live-keskkondades

SAP allikate loogika:

- **SAP TEST:**
 - kasutatakse ODS → EDW → DWH testkeskkonda laadimiseks
 - andmemahud on ligikaudu **3× väiksemad** kui live
 - andmed ei kasva orgaaniliselt; muutused toimuvad käsitsi või arendajate initsiatiivil
- **SAP LIVE:**
 - kasutatakse ainult live keskkonda laadimiseks
 - andmed arenevad loomulikult äriprotsessidest

Andmeid ei tooda keskkondade vahel üle.