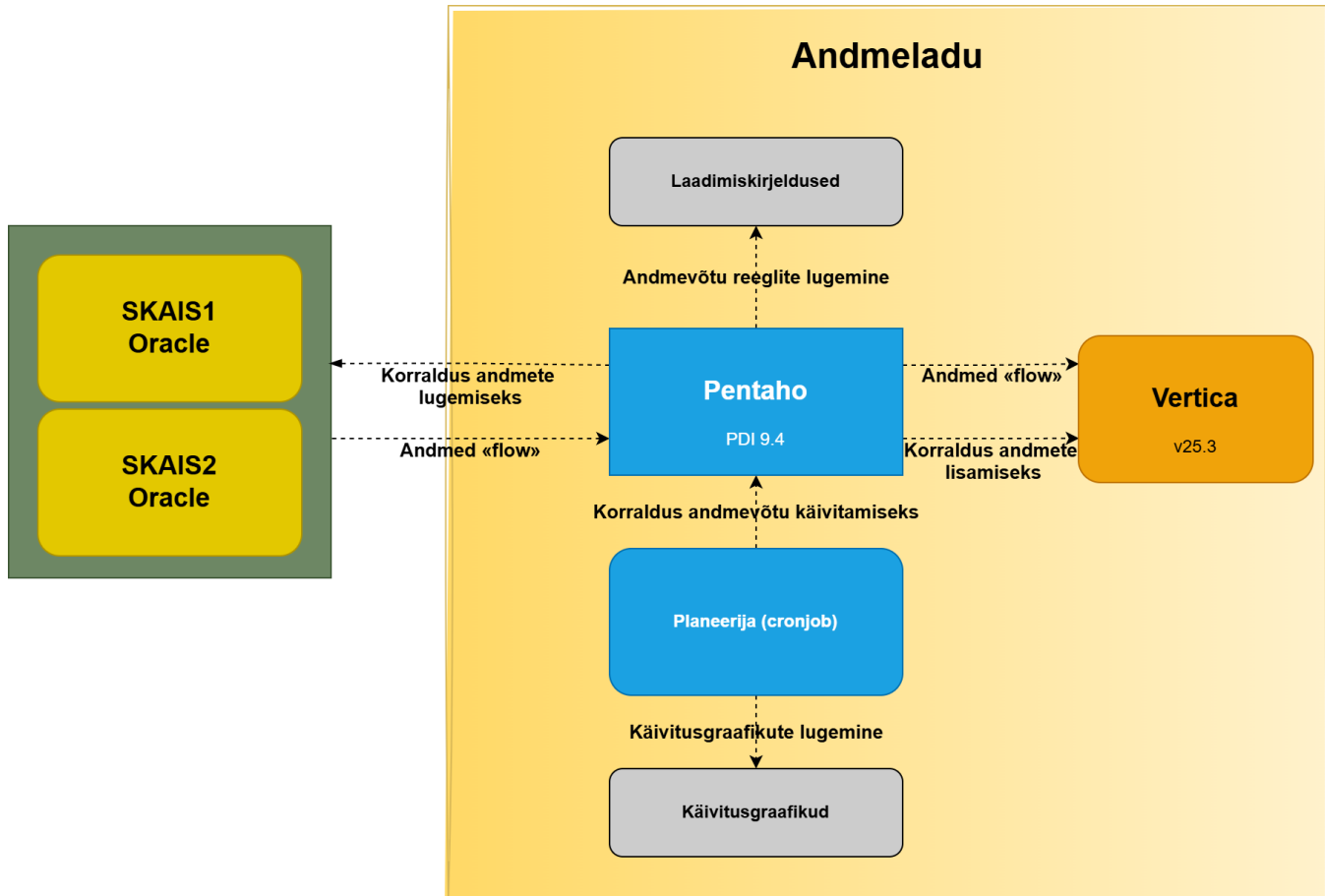


1. SKAIS andmeladu info	2
1.1 Serverid ja ühendused	4
1.2 Arendus	5
1.3 Automaatprotsesside käivitamine	6
1.4 Ligipääsud	7
1.5 SLA (arendustööde tingimused / rakenduste teenustasemed)	9
2. SKAIS1 andmeladu	10
2.1 SKAIS1 andmemudel	11
2.2 SKA DW andmebaas	12
2.3 SKA DW andmete laadimine	13
2.4 SKA DW süsteemsed tabelid	14
2.4.1 COLUMNS	15
2.4.2 DATA_SOURCE_PARAMS	16
2.4.3 DATA_SOURCES	17
2.4.4 DATA_TYPE_CONVERSIONS	18
2.4.5 DWH_SCHEMAS	19
2.4.6 EVENT_LOG	20
2.4.7 IMP_COLUMNS	21
2.4.8 IMP_SCHEMAS	22
2.4.9 IMP_TABLES	23
2.4.10 LOAD_LOG	24
2.4.11 SCHEMAS	25
2.4.12 TABLES	26
2.5 SKA DWH Pentaho Data-Intergration paigaldusjuhend	28
2.6 SKA DWH Vertica ühendumise juhend	31
2.7 Skeemid	34
2.8 Täislaadimiste tegemine	35
2.9 Logimine	36
3. SKAIS2 andmeladu	37
3.1 Laadimiste haldus	38
3.2 Logid	40
3.3 Pentaho PDI	41
3.4 SKA SKAIS2 DW andmebaas	42
3.5 SKA SKAIS2 DW andmete laadimine	43
3.6 SKA SKAIS2 DWH Pentaho Data-Intergration paigaldusjuhend	67
3.7 SKA SKAIS2 DWH Vertica ühendumise juhend	71
3.8 SKA SKAIS2 DW süsteemsed tabelid	74
3.8.1 COLUMNS SKAIS2	75
3.8.2 DATA_SOURCE_PARAMS SKAIS2	76
3.8.3 DATA_SOURCES SKAIS2	77
3.8.4 DATA_TYPE_CONVERSIONS SKAIS2	78
3.8.5 DMARTS SKAIS2	79
3.8.6 DWH_SCHEMAS SKAIS2	80
3.8.7 EVENT_LOG SKAIS2	81
3.8.8 IMP_COLUMNS SKAIS2	82
3.8.9 IMP_SCHEMAS SKAIS2	83
3.8.10 IMP_TABLES SKAIS2	84
3.8.11 LOAD_LOG SKAIS2	85
3.8.12 PARAMETERS SKAIS2	86
3.8.13 ROW_COUNTS SKAIS2	87
3.8.14 SCHEMAS SKAIS2	88
3.8.15 TABLES SKAIS2	89
3.9 Skeemid Skais2	91

SKAIS andmeladu info



SKAIS andmelaos laaditakse Skais1 ja Skais2 Oracle andmebaasidest andmed lao Vertica andmebaasi - andmelao kihtide skeemidesse.

Laadimist teostab (ühendused loob) Pentaho PDI, millele on ehitatud laadimismootor (PDI skriptid) Resta poolt.

Laadimismootor otsustab Verticas asuva info põhjal (ADM skeemid) - kust, mida, kuidas ja kuhu laadida ning tegeleb logimise, veahalduse ja teavitusega.

Tähtsaimad viited

- [Serverid ja ühendused](#)
- [Arendus](#)
- [Automaatprotsesside käivitamine](#)

SKAIS1

- [SKA DW andmebaas](#)
- [SKA DW andmete laadimine](#)
- [Logimine](#)
- [Skeemid](#)

SKAIS2

- [SKA SKAIS2 DW andmebaas](#)
- [SKA SKAIS2 DW andmete laadimine](#)
- [Logid](#)
- [Skeemid Skais2](#)

Kontaktid

- SKA äripoole kasutaja - Meelis Põlda.
- SKAIS2 Oracle kontakt TEHIK-us - RAIN
- Pentaho ja Vertica - [Matthias Johann Kurs](#)
- Graafikuid ja kirjeldusi koostab Resta OÜ.

Serverid ja ühendused

Test

Hetkel testkeskkond puudub ([SKANDL-82](#))

Live

- Pentaho DI ehk laadimisi käivitav server - pentaho-vertica-1a (10.11.25.27)
- Vertica ehk andmelao andmebaas - wrh-db.prd.tehik.ee (10.11.11.182)
Vertica andmebaasil on 4 node-i:

v_wrh_db_node0001	10.11.25.101
v_wrh_db_node0002	10.11.25.102
v_wrh_db_node0003	10.11.25.103
v_wrh_db_node0004	10.11.25.104

Andmebaasid

- Vertica (target Skais1 ja Skais2) - wrh-db.prd.tehik.ee:5433/wrh_db
Skais1 ja Skais2 andmeladu asub samas Vertica andmebaasis, eraldatud skeemides
- Oracle (source Skais2) - skais2db1.ml.ee:1521/SKAIS2APP
- Oracle (source Skais1) - skaprddb1.ml.ee:1521/SKAPRDAPP

Kasutajad (millega PDI loob ühendused baasidega)

- Skais1 Oracle ühendus - SKA_ANDMEAIT
- Skais2 Oracle ühendus - SKA_ANDMEAIT
- Skais1 Vertica ühendus - pentaho
- Skais2 Vertica ühendus - pentaho_skais2

Ühendustel vertica baasi kasutada **tlsmode=require**

Arendus

GIT

Arendus toimub Tehnik gitlab versioonihaldus platvormil

SKAIS1 PDI skriptid

https://gitlab.sotsiaalministeerium.ee/skandl/skais_dwh/skais1_pentaho.git

SKAIS2 PDI skriptid

https://gitlab.sotsiaalministeerium.ee/skandl/skais_dwh/skais2_pentaho.git

SKAIS2 Vertica skriptid ehk tarded

https://gitlab.sotsiaalministeerium.ee/skandl/skais_dwh/skais2_vertica.git

PDI skriptid

PDI skriptide (kjb, ktr, kdb) repositooriumi paigaldamisel jälgida, et nendes ei oleks ühenduste paroole (Encrypted *****)

Õigused

Uute objektide lisamisel (tabelid, vaated) lisab arendaja ka õigused. Täpsemalt [Ligipääsud](#)

Automaatprotsesside käivitamine

SKA andelao laadimised toimuvad ühe peamise Pentaho job faili kaudu, mida käivitatakse crontab olevast sh skriptis.

Hetkel laetakse SKAIS1 ja SKAIS2 andmebaasi andmelao vertica-sse wrh-db.prd.tehik.ee

SKAIS1 laadimise loogika on täpsemalt kirjeldatud lehel [SKA DW andmete laadimine](#)

SKAIS2 laadimise loogika on täpsemalt kirjeldatud lehel [SKA SKAIS2 DW andmete laadimine](#)

Ajastus

- Skais 1 laadimine algab iga öö kell 1:00. Kestab umbes 3h
- Skais 2 laadimine algab iga öö kell 2:00, kuid ei lae kuu viimasel päeval ning 1-8 kuupäev kaasaarvatud. Kestab umbes 2h30m

Crontab

Serveris pentaho-vertica-1a käivitatakse sh skript, mis omakorda käivitab PDI job.

Käivitatakse kasutaja pentaho alt

```
sudo su - pentaho
```

Lisatud crontab kirjed, kus kutstakse shelli skript välja:

```
crontab -e

##Skais1 live
00 01 * * * /home/pentaho/load_all_skais1_live.sh
##Skais2 live
00 02 * * * /home/pentaho/load_all_skais2_live.sh
```

Antud juhul käivitatakse Skais1 laadimine igaöiselt kell 1:00 ja Skais2 2:00.

Laadimise peatamiseks kommenteerida see välja crontab-is # märgiga

Laadimis skripti näide:

```
###Adjust if needed
PDI_DIR=/opt/etl/data-integration94
REPO=skais2_v2_live
SUBDIR=/dwh_loader
JOB=load_one_ds
LOGLVL=Basic
LOGDIR=/var/logs/pentaho
ADM_SCHEMA=ADM_SKAIS2
LOGNAME=load_all_skais2_live.log
#Lisaks kuupäeva ja logi roteerumise loogika
nohup $PDI_DIR/kitchen.sh -rep=$REPO -dir=$SUBDIR -job=$JOB -param:adm_schema=$ADM_SCHEMA -param:
is_dynamic_model=0 -level=$LOGLVL > $LOGDIR/$CDATE-$LOGNAME 2>&1 &
```

Enh käsitsi käivitades SKAIS2 puhul

```
nohup /opt/etl/data-integration94/kitchen.sh -rep=skais2_v2_live -dir=/dwh_loader -job=load_one_ds -param:adm_schema=ADM_SKAIS2 -param:
is_dynamic_model=0 -level=Basic > /var_logs_pentaho/2025-10-31-load_all_skais2_live.log 2>&1 &
```

Ligipääsud

Uued skeemid

Kui laadimine peaks toimuma uute skeemi, siis arendaja koostab nende lisamiseks itabi (itabi@tehik.ee) tellimuse (sh. lisab kirja CC reale TEHIK-u projektijuhil). Tellimuse kirjeldus allpool.

Selle tellimusega peaks loodama skeemid, vastavad skeeminimi_viewer ja skeeminimi_owner rollid ning antama õigused üldistele ska_owner, ska_viewer, ska

Uued tabelid

Uute tabelite lisamisel arendaja lisab õigused vastavale skeemi rollidele:

```
create table ab.abc( c int);
```

```
grant select on ab.abc to ab_viewer;
```

```
grant all on ab.abc to ab_owner;
```

Näiteks:

```
grant select on ACTIVITY_CATEGORY,DOMAIN_CATEGORY,EXP_DEVIATION_ICF_CATEGORY,LOG_ACTIVITY_CATEGORY,LOG_DOMAIN_CATEGORY,LOG_EXP_DEVIATION_ICF_CATEGORY
```

```
to DW_SKAIS_DOCTOR_ASSESSMENT_ODS_VIEWER;
```

Skais2 andmelao ligipääsude tellimine

Ligipääsud on skeemi põhised ning ligipääsud tellitakse skeemile.

Ligipääsud üldiselt tellitakse rollidele, mitte kasutajatele.

<https://gitlab.sotsiaalministeerium.ee/andmelaod/datawarehouse/ska>

Ska rollid

ska skeem on ska analüütikute tootearendus/liivakast skeem. Selleks, et ska analüütikud saaksid oma tabelleid ja objekte oma skeemi luua oleks tarvis neile anda rollid:

ska - teenuskonto(andmebaasi kasutaja) ja andmebaasi skeem

ska_viewer - roll, mis anna vaatmisõigused kõikidele skais1 ja skais2 skeemidele ska_owner - roll, mis anna vaatmisõigused kõikidele skais1 ja skais2 skeemidele ning objektide loomise õiguse ska skeemi

Selleks, et ska_viewer ja ska_owner rollid toimiksid peab iga kord kui luuakse uus ska skeem andma nimetatud rollidele loodud skeemidele SELECT ja USAGE õigused:

Näiteks DW_SKAIS_DOCUMENT_MANAGEMENT_ODS ja DW_SKAIS_DOCUMENT_MANAGEMENT_STG skeemide loomise järel tuleks andmebaasis käivitada:

```
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS_DOCUMENT_MANAGEMENT_ODS to ska_viewer, ska_owner, ska;  
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS_DOCUMENT_MANAGEMENT_STG to ska_viewer, ska_owner, ska;
```

Nimetatud koodijupp tuleks lisada skeemi loomise tellimuse TKT piletisse ning antud repositoorimi ska.sql faili. Täiendada feature harus, ning saata merge request Karl Kukk:

<https://gitlab.sotsiaalministeerium.ee/andmelaod/datawarehouse/ska/-/blob/master/ska.sql>

Tellimine

1. Täiendada ska.sql feature harus, ning saata merge request kellele?
2. Peale master jõudmist teha tellimus
 - a. Lisada SKAIS2 DWH live verticas skeemid - näiteks DW_SKAIS_DOCUMENT_MANAGEMENT_ODS, DW_SKAIS_DOCUMENT_MANAGEMENT_STG
 - b. Lisada rollide õigused antud skeemile lähtudes õiguste init failist <https://gitlab.sotsiaalministeerium.ee/andmelaod/datawarehouse/ska/-/blob/master/ska.sql>

```
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS_DOCUMENT_MANAGEMENT_ODS to ska_viewer, ska_owner, ska;  
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS_DOCUMENT_MANAGEMENT_STG to ska_viewer, ska_owner, ska;
```

3. Peale skeemi tegemist ja õiguste lisamist saab tabelid laadimisse lisada

Kogu tellimuse näide:

Palun lisada skeemid ja õigused Skais2 DWH live. Seoses task SKANDL-78

```
CREATE SCHEMA DW_SKAIS2VALINE_STG;  
CREATE SCHEMA DW_SKAIS2VALINE_ODS;  
GRANT ALL ON SCHEMA DW_SKAIS2VALINE_STG TO pentaho_skais2;  
GRANT ALL ON SCHEMA DW_SKAIS2VALINE_ODS TO pentaho_skais2;  
  
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS2VALINE_STG to ska_owner;  
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS2VALINE_ODS to ska_owner;  
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS2VALINE_STG to ska_viewer;  
GRANT SELECT,USAGE ON SCHEMA DW_SKAIS2VALINE_ODS to ska_viewer;
```

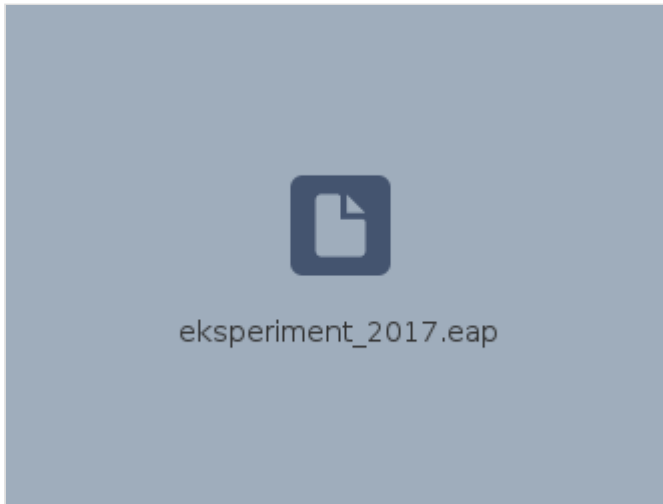
SLA (arendustööde tingimused / rakenduste teenustasemed)

SKA andmeladu

Teenuse parameeter	Väärtus
ISKE turvaklass	K1T1S2
Rakenduse tööaeg (millal laoplatvorm peab olema kättesaadav)	07:00 - 19:00
Kasutajatugi (teenindusaeg)	E-R 9.00-17.00
Planeeritud hooldusaeg	19.00 – 21.30 01.00 – 05.00
Minimaalne aeg teavitusest plaanilise katkestuseni	48 tundi
Maksimaalne plaanilise katkestuse kestvus	8 tundi
Maksimaalne plaaniliste katkestuste sagedus	8 korda kuus
Maksimaalne planeerimata katkestuste sagedus	8 korda kuus
Ühekordne maksimaalne planeerimata katkestuse kestus	8 tundi
Summaarne maksimaalne plaanimata katkestuste kestus kuus	16 tundi
Samaaegsete andmeanalüüsivahendite kasutajate arv	Kuni 40 kasutajat korraga
Samaaegsete aruannete kasutajate/päringute arv	Kuni 120 kasutajat korraga
Laadimise piirangud	Vaikimisi periood, millal SKAIS2 laadimisi ei tohi teostada on kuu viimasel päeval kuni järgmise kuu 8 esimest päeva. Seda perioodi võib Tellija muuta lähtuvalt ärivajadustest.

SKAIS1 andmeladu

SKAIS1 andmemudel



SKAIS1 andmemudel

SKA DW andmebaas

SKA DW andmebaasi (SKA andmelao) andmebaasisüsteemina kasutatakse spetsiaalselt analüütilisteks lahendusteks optimeeritud veerupõhist andmebaasimootorit OpenText Vertica.

SKA DW andmebaas jaguneb andmete laadimise loogika mõttes kahte kihti:

1. Eellaadimise ala (staging ehk STG-kiht) sisaldab andmete laadimise tööks vajalikke andmetabeleid (ajutised andmed). STG-kihi andmetabelite struktuur on täpselt sama, mis on lähtesüsteemi andmetabelitel (mõningased erinevused võivad tulla sellest, kuidas erinevad andmebaasisüsteemid (Oracle ja Vertica) salvestavad sarnaseid andmetüüpe nt varchar2(100) Oracles vs varchar(100) Verticas). Tüüpiliselt laaditakse igaõise laadimise käigus STG-kihti viimase ööpäeva jooksul lähtesüsteemidesse lisatud ja/või seal muudetud andmed. Erandjuhtumitel (kui lähtesüsteemis muudetud kirjade tuvastamine ei ole võimalik), teostatakse tabelile igal öösel täislaadimine. Eeskirjad, millist andmete laadimise meetodit ja see, kas vastava tabeli andmeid üldse SKAIS1 andmelattu laaditakse, on kirjeldatud süsteemses andmetabelis ADM.TABLES.
2. Operatiivne andmehoidla (Operational Data Store e ODS-kiht) sisaldab kõiki lähtesüsteemide andmebaaside andmeid (st kogu muudatuste ajalugu alates andmelao laadimise alghetkest). ODS kihi andmemudel on sarnane lähtesüsteemi andmemudelitele - igale lähtesüsteemi tabelile vastab ODS kihis samanimeline tabel, mis sisaldab kõiki vastava andmetabeli välju ning nelja täiendavat välja kirjade versioneerimiseks. Lisatavateks väljadeks on:
 - a. dwh_id - Surrogaatvõti (andmelao laadimisprogrammi poolt genereeritud võtmeväli)
 - b. is_valid - 0/1 väli, mis näitab, kas tegemist on kehtiva kirjega (kõige viimase versiooniga)
 - c. valid_from - versiooni kehtivuse algusaeg (kuupäev ja kellaeg)
 - d. valid_to - versiooni kehtivuse lõppaeg (kuupäev ja kellaeg)

Lisaks nimetatud kihtidele on eraldi skeem (nimega ADM_SKAIS1), mis sisaldab andmete laadimiseks ning muuks SKA DW toimimiseks vajalikke andmetabeleid.

SKA DW andmete laadimine

Andmete laadimiste üldine loogika

SKA_DW andmete laadimine toimub järgmise skeemi kohaselt

1. Lähtesüsteemi (SKAIS) andmemudeli muudatuste tuvastamine (sh SKA DW andmemudeli uuendamine)
 - a. Lähteallika (SKAIS) andmebaasi süsteemsetest tabelitest loetakse lähtesüsteemi andmemudel ning salvestatakse see süsteemsetesse tabelitesse ADM.IMP_SCHEMAS, ADM.IMP_TABLES ja ADM.IMP_COLUMNS
 - b. Lähteallika andmemudelit võrreldakse SKA DW poolt teada oleva mudeliga (st eelmisel päeval kehtinud andmemudeliga) ning tuvastatakse muudatused.
 - c. Kõikide nende andmetabelite korral, mis on laadimisse sisse lülitatud (LOAD_STATUS = 'LOAD' või 'NEW_LOAD') korral SKA DW andmemudelit (STG-tabeleid ja ODS-tabeleid) muudetakse vastavalt eelmises punktis tuvastatud muudatustele (lisatakse uued väljad, muudetakse olemasolevate väljade tüüpe, lähtesüsteemist kustutatud väljade laadimine lülitatakse välja (LOAD_STATUS = 'NO_OBJECT')).
2. Andmete import ja andmete laadimine STG ja ODS tabelitesse (iga tabeli korral võidakse kasutada erinevat andmete impordi meetodit, erinevat lähtesüsteemist kustutatud kirjade tühistatuse märkimise meetodit jms Vt tabelite laadimise konfigureerimise võimalusi süsteemsete tabelite [ADM_SKAIS1.TABLES](#) ja [ADM_SKAIS1.COLUMNNS](#) kirjelduste juurest).
 - a. (Pärast viimast laadimist lisatud ja/või muudetud) andmete import STG tabelitesse (sh viimati laaditud andmete tuvastamine kasutades ADM.TABLES.LAST_LOAD_EXPRESSION avaldist)
 - b. Andmete laadimine ODS tabelitesse (sh lähtesüsteemist kustutatud kirjade märkimine kehtetuks)
 - c. Lähtesüsteemi ja ODS tabeli kirjade arvu võrdlemine.

Andmelao laadimiste kohta tekib laadimiste logi. Laadimisprotsesside käivitamise ja lõpetamise teated salvestatakse laadimise logi tabelisse [LOAD_LOG](#), detailsemad tabelite loomise (sündmuste teated) salvestatakse tabelisse [EVENT_LOG](#).

Laadimiste käitumine veaolukordades

Laadimise alguses kontrollitakse, kas eelmine (sama andmeallika) laadimine on lõppenud. Kui eelmine laadimise protsess ei ole lõppenud või lõppes veaga ning ei ole märgitud, et viga tuleb ignoreerida, siis järgmist sama allika laadimisprotsessi ei käivitata.

Kui eelmise päeva laadimine sai vea ning vigane olukord on parandatud (st järgmisel päeval saab laadimist uuesti jätkata), siis tuleb laadimiste logi tabelis [IS_IGNORE](#) väärtuseks panna 1 (st viga ignoreeritakse).

Juhul, kui mingi STG tabeli laadimise käigus tekib viga, siis STG tabelite laadimine töötab lõpuni, samuti käivitatakse ODS tabelite laadimine kuid vea saanud tabeli korral vastava ODS tabeli laadimist ei käivitata.

Laadimiste tulemustest teavitamine

Laadimiste tulemuste teavitamiseks saadetakse e-mail aadressile (vastava e-mail'i aliase loob ning selle aliase taga paiknevaid konkreetseid e-mail'i aadresse haldab TEHIK, e-mail'i alias tuleb sisestada tabelisse PARAMETERS parameetri to_email väärtuseks).

Juhul, kui laadimiste käigus ei tekkinud vigu ega hoiatusi, saadetakse e-mail teemaga "Andmeallika SKAIS1 laadimine õnnestus".

Juhul, kui laadimise käigus tekkis hoiatusi, saadetakse e-mail teemaga "Andmeallika SKAIS1 laadimisel esines hoiatusi!".

Juhul, kui laadimise käigus tekkis vigu, saadetakse e-mail teemaga "Andmeallika SKAIS1 laadimine sai vigu!" ning e-mail'i sisuks on vigade logi.

Juhul, kui tavapäraseks laadimise lõppemise ajaks ei ole e-mail'i saanud, siis tuleb olukorda käsitleda veaolukorrana ning täpsemat infot on võimalik leida logitabelitest ja/või Pentaho logist.

SKA DW süsteemsed tabelid

COLUMNS

Väljade/veergude andmed

Välja nimi	Välja tüüp	Selgitus	Andmete näited
COLUMN_VERSION_ID	int	Surrogaatvõti	55
DATA_SOURCE_CODE	varchar (20)	Andmeallika/lähtesüsteemi kood	'SKAIS'
SOURCE_SCHEMA_NAME	varchar (128)	Skeemi nimi lähtesüsteemis	'SKA_OWNER'
SOURCE_TABLE_NAME	varchar (128)	Tabeli nimi lähtesüsteemis	'ADDRESS'
SOURCE_COLUMN_NAME	varchar (128)	Välja/veeru nimi lähtesüsteemis	'ADDRESS_KOOD'
DWH_COLUMN_NAME	varchar (128)	Välja/veeru nimi andmelaos	'ADDRESS_KOOD'
SOURCE_COLUMN_COMMENT	varchar (4000)	Välja/veeru kirjeldus lähtesüsteemis	'Kommentaar'
DWH_COLUMN_COMMENT	varchar (4000)	Välja/veeru kirjeldus andmelaos	'Kommentaar'
DATA_TYPE	varchar (128)	Andmetüüp lähtesüsteemis	'VARCHAR2', 'INTEGER', 'DECIMAL'
DATA_LENGTH	int	Andmevälja pikkus lähtesüsteemis	1020
DATA_PRECISION	int	Andmevälja täpsus lähtesüsteemis	18
DATA_SCALE	int	Andmevälja täpsus peale koma lähtesüsteemis	0
DWH_DATA_TYPE	varchar (255)	Andmetüüp andmelaos	'VARCHAR(1020)'
NULLABLE	char(1)	Kas veerg võib sisaldada puuduvaid (NULL) väärtusi (0/1)	1
IS_KEY	int	Kas veerg on primaarvõti (0/1)	0
LOAD_STATUS	varchar (20)	Välja/veeru laadimise staatus andmelaos	'LOAD' - Välja laadimine on sisse lülitatud 'NEW_LOAD' - Tegemist on uue väljaga (st haldur pole välja laadimise staatust üle vaadanud /kinnitanud), väli on laadimisse sisse lülitatud 'NEW_NOT_LOAD' - Tegemist on uue väljaga, väli ei ole laadimisse sisse lülitatud 'NOT_LOAD' - välja laadimine ei ole sisse lülitatud 'NO_OBJECT' - välja ei ole enam lähtesüsteemis (varem oli)
DWH_COLUMN_STATUS	varchar (30)	Välja staatus andmelaos:	'NEEDS_CREATION' - väli vajab loomist 'NEEDS_MODIFICATION' - väli vajab muutmist (näiteks lähteallikas on muutunud andmetüüp) 'READY' - väli on loodud/muudetud
SOURCE_COLUMN_ID	int	Välja/veeru ID lähtesüsteemis	11
IS_VALID	int	Kas tegemist on kehtiva kirje /versiooniga (0/1)	1
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2018-05-04 17:26:58'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2018-05-04 11:22:58'
MODIFIED_BY	varchar (32)	Kirje looja/muutja	'pentaho'

DATA_SOURCE_PARAMS

Lähtesüsteemi andmebaasi parameetrid

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(20)	Andmeallika/lähtesüsteemi kood	'SKAIS'
DB_SERVER_NAME	varchar(32)	Serveri nimi või IP aadress	'test-test.ml.ee '
DB_PORT_NR	int	Pordi number	1234
DB_NAME	varchar(32)	Andmebaasi nimi	'TSKAIS'
DB_USERNAME	varchar(32)	Kasutajanimi	'SKAIS_USER_NAME'
DB_PASSWORD	varchar(128)	Parooli räsi (luuakse pentaho encr.sh abil). Vt täpsemalt paigaldusjuhendist.	'Encrypted 0123456789ABCDEF'

DATA_SOURCES

Andmeallikad (lähte-andmebaasid)

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_VERSION_ID	int	Andmeallika versiooni ID. Surrogaativõti.	1
DATA_SOURCE_CODE	varchar (32)	Andmeallika/lähtesüsteemi kood.	'SKAIS'
DATA_SOURCE_NAME	varchar (128)	Lähtesüsteemi nimetus	'Töötukassa aruandlusmoodul'
DATA_SOURCE_TYPE	varchar (20)	Lähtesüsteemi tüüp	'SOURCE_DB' - lähtesüsteemiks on andmebaas
DBMS_TYPE	varchar (20)	Andmebaasi tüüp	'ORACLE'
IS_ACTIVE	int	Kas lähtesüsteemist andmete laadimine on aktiivne (võimaldab terve lähtesüsteemi laadimise korraga välja lülitada)	1
IS_LOAD_TABLES	int	Kas lähtesüsteemist laaditakse andmetabeleid (table)	0
IS_LOAD_VIEWS	int	Kas lähtesüsteemist laaditakse vaateid (view)	0
IS_LOAD_MVIEWS	int	Kas lähtesüsteemist laaditakse materialiseeritud vaateis (materialized view)	1
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	1
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2018-05-04 17:21:26'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2018-05-04 17:55:00', NULL
MODIFIED_BY	varchar (32)	Versiooni lisaja/muutja	'SYSTEM'

DATA_TYPE_CONVERSIONS

Välja tüüpide teisendustabel

Välja nimi	Välja tüüp	Selgitus	Andmete näited
ID	int	Surrogaatvõti	2
DBMS_TYPE	varchar(20)	Lähtesüsteemi andmebaasi tüüp	'ORACLE'
DATA_TYPE	varchar(128)	Andmetüüp lähtesüsteemis	'DATE'
DATA_LENGTH	int	Andmevälja pikkus lähtesüsteemis	10
DATA_PRECISION	int	Andmevälja täpsus lähtesüsteemis	1
DATA_SCALE	int	Andmevälja täpsus peale koma lähtesüsteemis	1
TARGET_DATA_TYPE	varchar(128)	Andmetüüp (sh pikkus ja täpsus) andmelaos	'DATETIME'

DWH_SCHEMAS

Andmelao skeemide andmed.

Iga siin kirjeldatus skeemi kohta luuakse andmelaos vähemalt 2 skeemi (<skeemi nimi>_STG - vastava loogilise andmeallika *staging* tabelite skeem, <skeemi nimi>_ODS - vastava loogilise andmeallika ODS tabelite skeem). Lisaks luuakse vajaduse korral (kui HAS_DMARTS = 1) skeem <skeemi nimi>_DMART, mis sisaldab vastava andmeallika dimensiooni- ja faktitabeleid ja andmelette.

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DWH_SCHEMA_VERSION_ID	int	Skeemi versiooni ID	1
DWH_SCHEMA_NAME	varchar(32)	Andmelao skeemi/loogilise andmeallika nimi	'ORACLE'
DWH_SCHEMA_COMMENT	varchar(255)	Andmelao skeemi kirjeldus	'Töötute registreerimise, tööturuteenuste ja -toetuste menetlemise infosüsteem'
DWH_SCHEMA_STATUS	varchar(30)	Andmelao skeemi staatus	'NEEDS_CREATION' - skeem on loomata 'NEEDS_MODIFICATION' - skeemis on objekte (tabeleid), mida on vaja muuta 'READY' - skeem ja kõik seal olevad objektid on loodud/muudetud
HAS_DMARTS	int	Kas skeem sisaldab andmelette	1
IS_ACTIVE	int	Kas skeemi/loogilise andmeallika andmete laadimine on aktiivne.	1
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	0
VALID_FROM	Timestamp	Kirje/versiooni kehtivuse algusaeg	'2018-05-04 17:21:26'
VALID_TO	Timestamp	Kirje/versiooni kehtivuse lõppaeg	'2018-05-04 17:26:57'
MODIFIED_BY	varchar(32)	Versiooni lisaja/muutja	'pentaho'

EVENT_LOG

Sündmuste logi

Välja nimi	Välja tüüp	Selgitus	Andmete näited
EVENT_ID	int	Sündmuse ID. Primaarvõti.	1
EVENT_DT	Timestamp	Sündmuse toimumise/registreerimise kuupäev ja kellaaeg	'2018-05-04 17:21:25'
EVENT_CLASS	varchar(20)	Sündmuse grupi kood. Võimalikud väärtused: INSTALLATION -	'INSTALLATION', 'LOADING'
EVENT_TYPE	varchar(20)	Sündmuse tüüp. Võimalikud variandid: ERROR - viga, WARNING - hoiatus, SUCCESS - edukas tegevus, INFO - teade	'ERROR'
EVENT_MESSAGE	varchar(255)	Sündmuse kirjeldus	'ERROR in loading STG table'
RELATED_OBJECT	varchar(50)	Sündmusega seotud objekt (nt tabeli nimi, lähteallika nimi vms). Konkreetne objekt sõltub sündmuse koodist.	'ADM.TABLES'
PROCESS_NAME	varchar(50)	Pentaho protsessi nimi, mis sündmuse tekitas	'run_upgrade'
STEP_NAME	varchar(50)	Pentaho sammu (step) nimi, mis sündmuse tekitas	'schema & main tables'

IMP_COLUMNS

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(32)	Lähtesüsteemi kood	'SKAIS'
SCHEMA_NAME	varchar(128)	Välja/veeru nimi lähtesüsteemis	'SKA_OWNER'
TABLE_NAME	varchar(128)	Tabeli nimi lähtesüsteemis	'ADDRESSID'
COLUMN_NAME	varchar(128)	Välja/veeru nimi andmelaos	'ADDRESS_KOOD'
COLUMN_COMMENT	varchar(4000)	Välja/veeru kirjeldus lähtesüsteemis	'Kommentaar'
DWH_COLUMN_COMMENT	varchar(4000)	Välja/veeru kirjeldus andmelaos	'Kommentaar'
DATA_TYPE	int	Andmetüüp lähtesüsteemis	'VARCHAR2'
DATA_LENGTH	int	Andmevälja pikkus lähtesüsteemis	1020
DATA_PRECISION	int	Andmevälja täpsus lähtesüsteemis	18
DATA_SCALE	int	Andmevälja täpsus peale koma lähtesüsteemis	0
NULLABLE	char(1)	Kas puuduvad (NULL) väärtused on lubatud	'Y'
IS_KEY	int	Kas väli/veerg on primaarvõti lähtesüsteemis	0
COLUMN_ID	int	Välja/veeru identifikaator lähtesüsteemis	18

IMP_SCHEMAS

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(32)	Andmeallika/lähtesüsteemi kood. Viide tabelisse DATA_SOURCES	'SKAIS'
SCHEMA_NAME	varchar(128)	Skeemi nimi lähtesüsteemis	'SKA_OWNER'

IMP_TABLES

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(32)	Andmeallika/lätesüsteemi kood. Viide tabelisse DATA_SOURCES	'SKAIS'
SCHEMA_NAME	varchar(128)	Skeemi nimi lähtesüsteemis	'SKA_OWNER'
TABLE_NAME	varchar(128)	Tabeli nimi lähtesüsteemis	'AADRESSID'
TABLE_TYPE	varchar(20)	Tabeli tüüp (TABLE - tabel, VIEW - vaade, MVIEW - materialiseeritud vaade)	'TABLE'
TABLE_COMMENT	varchar(4000)	Tabeli kirjeldus lähtesüsteemis	"

LOAD_LOG

Andmete laadimise logi

Välja nimi	Välja tüüp	Selgitus	Kommentaar	Andmete näide
LOG_ID	int	Logikirje identifikaator. Primaarvõti.		
PROCES_ID	int	Laadimise protsessi identifikaator.	Iga käivitatud andmete laadimise protsess saab unikaalse identifikaatori.	1
SUBPROCESS_CODE	varchar (50)	Alamprotsessi kood /nimi	Võimalikud väärtused: DATA_LOAD - andmete laadimise põhiprotsess DMART_LOAD - andmelettide laadimise protsess, ODS_LOAD - ODS tabelite laadimise protsess STG_LOAD - STG tabelite laadimise protsess COMPARE_ROW_COUNTS - kirjete arvu võrdlemise protsess COMPARE_DATA_MODEL - andmemudeli võrdlemise protsess UPDATE_DATA_MODEL - andmemudeli muutmise protsess	
DATA_SOURCE_CODE	varchar (32)	Andmeallika kood. Viide tabelile ADM.DATA_SOURCES		'SKAIS'
RELATED_OBJECT	varchar (50)	Protsessiga seotud objekti (skeemi, andmetabeli vms) kood		'DW_SKAIS_ODS.AADDRESSID'
START_DT	datetime	Protsessi käivitamise algusaeg		'2018-05-20 05:30:00'
END_DT	datetime	Protsessi käivitamise lõppaeg		'2018-05-20 06:30:28'
STATUS	varchar (32)	Laadimise protsessi (lõpetamise) staatus.	Võimalikud väärtused: STARTED - protsessi on alustatud aga see pole veel lõppenud SUCCESS - protsess on lõppenud edukalt WARNING - protsess on lõppenud hoiatusega, konkreetsed hoiatusteated asuvad tabelis EVENT_LOG ERROR - protsess on lõppenud veaga, konkreetsed veateated asuvad tabelis EVENT_LOG	
IS_IGNORE	int	Kas lõpetamise staatust võib ignoreerida.	Kui eelmine laadimise protsess ei ole lõppenud (on staatuses STARTED) või lõppes veaga (on staatuses ERROR), siis üldjuhul järgmist sama allika laadimisprotsessi ei käivitata. Kui on soov järgmine laadimine ikkagi käivitada, siis tuleb eelmisel laadimisprotsessil märkida IS_IGNORE = 1.	0

SCHEMAS

Lähtesüsteemi skeemid

Välja nimi	Välja tüüp	Selgitus	Andmete näited
HEMA_VERSION_ID	int	Skeemi versiooni ID	1
DATA_SOURCE_CODE	varchar(32)	Andmeallika/lähtesüsteemi kood. Viide tabelile DATA_SOURCES	'SKAIS'
SOURCE_SCHEMA_NAME	varchar(128)	Skeemi nimi lähtesüsteemis	'SKA_OWNER'
DWH_SCHEMA_NAME	varchar(32)	Skeemi nimi andmelaos	'DW_SKA_OWNER'
STATUS	varchar(20)	Skeemi andmete laadimise staatus	'LOAD'
IS_ACTIVE	int	Kas skeemi andmete laadimine on aktiivne (võimaldab terve skeemi andmete laadimise lihtsamalt välja lülitada) Hetkel ei kasutata	1
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	1
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2018-05-04 17:24:53'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2018-05-05 17:24:53'
MODIFIED_BY	varchar(32)	Versiooni lisaja/muutja	'pentaho'

TABLES

Tabelite andmed.

Välja nimi	Välja tüüp	Selgitus	Andmete näited
TABLE_VERSION_ID	int	Surrogaatvõti	281
DATA_SOURCE_CODE	varchar (32)	Andmeallika/lähtesüsteemi kood	'SKAIS'
SOURCE_SCHEMA_NAME	varchar (128)	Skeemi nimi lähtesüsteemis	'SKA_OWNER'
SOURCE_TABLE_NAME	varchar (128)	Tabeli nimi lähtesüsteemis	'AADRESSID'
DWH_SCHEMA_NAME	varchar (128)	Skeemi nimi andmelaos	'DW_SKA_OWNER'
DWH_TABLE_NAME	varchar (128)	Tabeli nimi andmelaos	'AADRESSID'
TABLE_TYPE	varchar (20)	Tabeli tüüp (TABLE, VIEW, MVIEW)	'TABLE'
SOURCE_TABLE_COMMENT	varchar (4000)	Tabeli kirjeldus lähtesüsteemis	"
DWH_TABLE_COMMENT	varchar (4000)	Tabeli kirjeldus andmelaos	"
LOAD_STATUS	varchar (20)	Laadimise staatus	'LOAD' - Tabeli laadimine on sisse lülitatud 'NEW_LOAD' - Tegemist on uue andmetabeliga (st haldur pole tabeli laadimise staatust üle vaadanud/kinnitanud), tabel on laadimisse sisse lülitatud 'NEW_NOT_LOAD' - Tegemist on uue andmetabeliga, tabel ei ole laadimisse sisse lülitatud 'NOT_LOAD' - tabeli laadimine ei ole sisse lülitatud 'NO_OBJECT' - tabelit ei ole enam lähtesüsteemis (varem oli)
LOAD_METHOD	varchar (20)	Andmete laadimise meetod	'LOAD_MODIFIED' - laaditakse pärast viimast õnnestunud andmete laadimist lisatud /muudetud andmed 'REWRITE' - andmed kirjutatakse üle (st muudatuste ajalugu ei säilitata) 'LOAD_MODIFIED_WOID' - laaditakse pärast viimast õnnestunud andmete laadimist lisatud andmed, tabelil puudub võtmeväli st uued andmed lisatakse, kirjete muutmist ei arvestata (vanade versioonide kehtetuks tunnistamist ei ole).
MODIFIED_FILTER	varchar (255)	Filtritingimus, mida kasutatakse muudetud ja/või lisatud andmete tuvastamiseks (LOAD_MODIFIED) laadimismeetodi korra.	'SYS_MUUTMISE_AEG >=TO_DATE(' LAST_LOAD_DT ',' yyyy-mm-dd hh24:mi:ss ')
LAST_LOAD_EXPIRATION	varchar (255)	Avaldis, mis tuvastab mis aja seisuga andmed on laaditud	'timestampadd('hour',-1, max(sys_muutmise_aeg))'
DWH_TABLE_STATUS	varchar (30)	Tabeli staatus andmelaos	'NEEDS_CREATION' - tabel vajab loomist 'NEEDS_MODIFICATION' - tabel vajab muutmist 'READY' - tabel on loodud/muudetud
LAST_LOAD_DT	timestamp	Viimase õnnestunud andmete laadimise kuupäev ja kellaaeg	'2018-05-15 00:01:10'
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	0
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2018-05-04 17:24:53'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2018-05-22 21:43:07'
MODIFIED_BY	varchar (32)	Versiooni lisaja/muutja	'pentaho'

IS_DM_TABLE	int	Kas tegemist on andmelettide arvutamisel kasutatava andmetabeliga. 0/1 tunnus.	Kui mõne andmelettide laadimises kasutatava tabeli laadimisel tekib viga, siis andmelettide laadimise osa ei käivitata.
IS_ROW_COUNT	int	Kas tabeli laadimisel võrreldakse andmelattu jõudnud kirjete arvu ja lähteallikas olevat kirjete arvu. 0/1 tunnus	
PRIORITY	int	Andmetabeli laadimise prioriteetsus. Väiksema väärtusega andmetabelid laaditakse enne.	
OBJECT_ID	int	Objekti (tabeli) ID.	See viitab tabelites XXX_LOG, XXX_LISALOG ja HISTORID kasutatavale object_id-le. Võimaldab leida vastava tabeli logikirjeid.
DELETED_STMT	varchar (255)	Kirjete kustutamise update lause	<p>Üldjuhul kasutatakse lähtesüsteemist kustutatud kirjete märkimiseks järgmist SQL lauset:</p> <pre>update <tabel> t1 set is_valid = 2, valid_to = t2.log_kuup from (select n1, log_kuup from <deleted_log_table> where log_op = 'D' and log_obj_id = <object_id>) t2 where <key_column_name> = t2.n1 and t1.is_valid = 1</pre> <p>Juhul kui standartne kustutatud kirjete kehtetuks märkimise lause ei sobi, siis saab siia välja kirjutada vajaliku update lause.</p>
DELETED_LOG_TABLE	varchar (255)	Millisest logitabelist tuleb võtta vastava tabeli kustutatud kirjete info.	Võimalikud variandid on XXX_LOG, XXX_LISALOG, HISTORID. Kui väärtus on tühi, siis vaikeväärtusena kasutatakse XXX_LOG.

SKA DWH Pentaho Data-Intergration paigaldusjuhend

Pigem kasutada uuemat juhendit siit: [Pentaho PDI](#)

Pentaho Data Integration on ETL tarkvara – andmete eksport, töötlemine ja ümberlaadimine.

Paigaldatakse Pentaho Community Edition (CE), mis on vabavaraline tarkvara

Kasutaja

Pentaho java protsess käivitatakse pentaho kasutaja alt.

```
sudo useradd pentaho
```

```
sudo passwd pentaho
```

Java JDK

Pentaho jaoks paigaldatakse uusim JDK8 versioon

RPM installi puhul paigaldatakse JDK asukohta /usr/java/default

```
cd /tmp/java
```

```
sudo rpm -Uvh jdk-8u151-linux-x64.rpm
```

Lisada JAVA_HOME keskkonnamuutuja pentaho kasutajale

```
vi /home/pentaho/.bash_profile
```

```
export JAVA_HOME=/usr/java/default/jre
```

Install

PDI ei vaja eraldi paigaldust. Arhiiv lahti pakkida ja õigused anda.

Pentaho Data Integration CE versiooni saab alla laadida sourceforge-st:

<https://sourceforge.net/projects/pentaho/files/Data%20Integration/>

Pentaho alla laadida, lahti pakkida ja õigused sättida.

```
cd /tmp/pentaho
```

```
unzip pdi-ce-8.0.zip
```

```
sudo mkdir /opt/data-integration
```

```
sudo mv * /opt/data-integration
```

```
cd /opt/data-integration
```

```
sudo chmod -R u+x *.sh
```

```
sudo chown -R pentaho *
```

JDBC andmeühendused

Pentaho on javal baseeruv ja seega kasutab andmebaasi ühendusteks JDBC ühendusi.

Pentaho ühendub oracle ja vertica andmebaasiga.

Andmebaasi draiverid

Paigaldada vertica ja oracle sobilikud andmebaasi draiverid.

Pentaho nimekiri draiveritest:

<https://help.pentaho.com/Documentation/5.3/0D0/160/010>

Vertica 9.0.0 draiveri leiab:

<https://my.vertica.com/download/vertica/client-drivers/>

Oracle draiveri versioon valida vastavalt andmebaasimootori versioonile

<http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html>

Draiverite jar failid liigutada pentaho lib kausta

`/opt/pentaho/data-integration/lib`

Pentaho repositoorium

Kirjeldada pentaho repositoorium kus hakkavad laadimiskriptid asuma

vi `/home/pentaho/.kettle/repositories.xml`

Jälgida asukohta – asukoht on osades skriptides sisse kirjutatud

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<repositories>
```

```
  <repository>  <id>KettleFileRepository</id>
```

```
    <name>ska</name>
```

```
    <description>SKA DWH REPO</description>
```

```
    <is_default>>false</is_default>
```

```
    <base_directory>/home/pentaho/repo</base_directory>
```

```
    <read_only>N</read_only>
```

```
    <hides_hidden_files>N</hides_hidden_files>
```

```
  </repository> </repositories>
```

Luaa repositooriumi kaust

```
sudo mkdir /home/pentaho/repo
```

```
sudo chown pentaho:pentaho /home/pentaho/repo
```

Repo olemasolu saab kontrollida:

```
cd /srv/pentaho
```

```
./kitchen.sh -listrep
```

Laadimiskriptid

Kopeerida pentaho failid (laadimiskriptid) pentaho repositooriumi kausta.

```
cp -R /tmp/laadimiskriptid/* /home/pentaho/repo
```

```
sudo chown -R pentaho:pentaho /home/pentaho/repo
```

Andmeühenduste failid

Pentaho repositooriumis asuvad andmeühendusi defineerivad kdb failid. Ühenduse failides tuleb muuta parameetrid live Vertica-le vastavaks Krüpteerida vertica kasutaja „pentaho“ parool ja SKAIS andmebaasi ühenduse kasutaja. Salvesta väljund, mis on kujul „Encrypted 123456...“

```
sudo su - pentaho
```

```
/opt/data-integration/encr.sh -kettle vertica_pentaho_parool
```

Conn_ska_wrh on vertica ühendust kirjeldav fail

```
vi /home/pentaho/repo/conn_ska_wrh.kdb
```

Muuta vertica ühendusel serverinimi, baasinimi, port, parool(pentaho Vertica kasutaja). Port on kirjeldatud kahes kohas. Nimi conn_ska_wrh peab samaks jääma

```
<connection>
  <name>conn_ska_wrh</name>
  <server>ska-wrh-1a.prd.tehik.ee</server>
  <type>VERTICA5</type>
  <access>Native</access>
  <database>ska_wrh_db</database>
  <port>5433</port>
  <username>pentaho</username>
  <password>Encrypted *****</password>
  <servername/>
  <data_tablespace/>
  <index_tablespace/>
  <attributes>
.....
  <attribute><code>PORT_NUMBER</code><attribute>5433</attribute></attribute>
```

Pentaho skriptide jooksutamine

Pentaho skripte on manuaalselt võimalik jooksutada pentaho serveris.

rep määrab repositooriumi nime. dir määrab alamkataloogi repositooriumis. job määrab pentaho job faili nime. Pikemate laadimiste puhul kasutada nohup &

```
sudo su - pentaho
```

```
nohup /opt/data-integration/kitchen.sh -rep=ska2 -dir=/load_ods -job=load_one_ds -level=Normal > /tmp/prelive1.log &
```

SKA DWH Vertica ühendumise juhend

Kasutaja

Skais vertica kasutajaid haldab Tehik (itabi)

JDBC

Java põhiste klientide kaudu ühendumine Dbeaver näitel

Draiver

Laadi alla Vertica JDBC uusima versiooni draiveri fail ja paiguta oma arvutis kindlasse kohta.

<https://www.vertica.com/download/vertica/client-drivers/>

https://www.vertica.com/client_drivers/24.2.x/24.2.0-0/vertica-jdbc-24.2.0-0.jar

Dbeaver

Laadi alla ja paigalda Dbeaver Community edition (koos JRE-ga)

<https://dbeaver.io/download/>

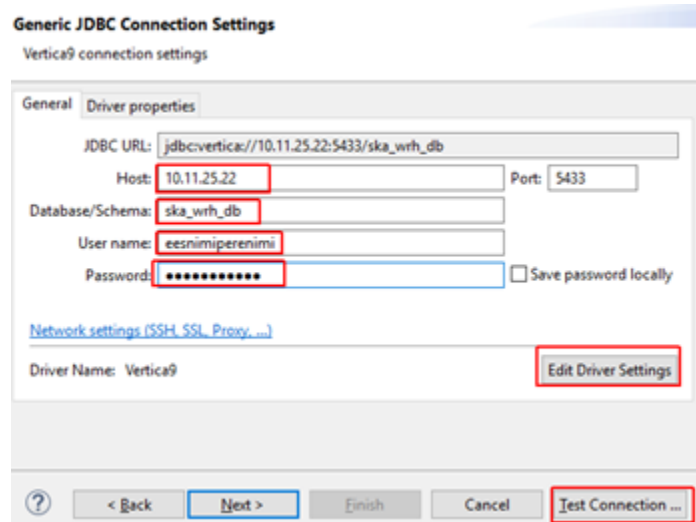
Keskkondade parameetrid

Vaata parameetreid lehelt [Serverid ja ühendused](#)

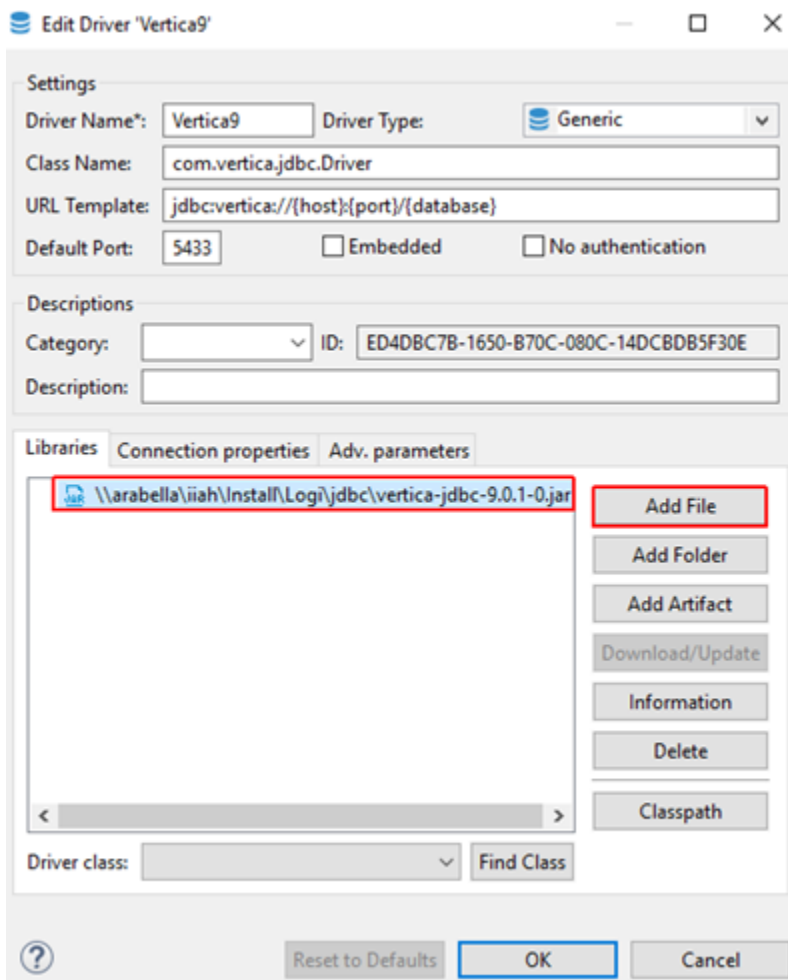
Dbeaver ühendus

Kui dbeaver on käivitatud, siis tuleb luua uus ühendus, määrata draiver ja sättida parameetrid.

New Connection > Connection Type Vertica > Määrata ühenduse parameetrid



Edit Driver Settings alt lisada eelnevalt allalaetud vertica-jdbc-***.jar fail



Test connection

Ühendu lisatud ühenduse andmebaasiga

ODBC

ODBC põhiste klientide kaudu ühendamine

Client

Tõmba alla ja paigalda Vertica uusim Windows Client

https://www.vertica.com/client_drivers/24.2.x/24.2.0-0/VerticaSetup-24.2.0-0.exe

ODBC

Lisada ODBC ühendus vastava baasi parameetritega

Start > ODBC Data Sources (64-bit) > Add > Vertica

Määra Database, Server, Port ja kasutaja Basic Settings alt.

HPE Vertica ODBC DSN Configuration

Basic Settings Client Settings Server Settings About

Connection info

DSN name *:

Description:

Database *:

Server *:

Backup servers:

Port:

Use connection load balancing:

Authentication

User name *:

Password:

Prompt for missing password:

Use Windows authentication:

Kasuta antud ODBC ühendust ODBC toetavas kliendis nagu Excel või mõni teine

Skeemid

Skeemi nimi	Kirjeldus	Staatus
ADM_SKAIS1	Admin tabelid - laadimiste sätted ja logid	
ADM_SKAIS1_V2		Vana, pole kasutuses
DW_AES_OWNER_ODS	AES Owner ODS	
DW_AES_OWNER_STG	AES Owner STG	
DW_SKA_OWNER_KEYS_ODS	SKA OWNER KEYS ODS	
DW_SKA_OWNER_KEYS_STG	SKA OWNER KEYS STG	
DW_SKA_OWNER_ODS	SKA OWNER ODS	
DW_SKA_OWNER_STG	SKA OWNER STG	
TEST_SKAIS1		Tühi?

Täislaadimiste tegemine

Kuna SKAIS andmebaasis lülitatakse pikemate protsesside tegemisel ajaloo tabeli (HISTORID) täitmine välja, siis on pärast selliseid protsesse vajalik teha nimetatud tabelite täislaadimine.

Täislaadimise vajadust näitab see, kui kirjete arv lähtebaasis (SKAIS andmebaas) ja andmelaos vastavas tabelis on erinev. Laadimise käigus võrreldakse kirjete arvu lähtebaasis ja andmelaos ning tulemus salvestatakse tabelisse ADM.ROW_COUNTS. Välja STATUS_CODE väärtus WARNING näitab, et lähtesüsteemi tabelis ja andmelaos on erinev kirjete arv; veerud SOURCE_COUNT ja DWH_COUNT näitavad vastavalt kirjete arvu lähtesüsteemis ja andmelaos.

NB! Kuna suuremate tabelite täislaadimine võib võtta mitmeid tunde (200 miljonit kirjet võtab üle 10 tunni), siis soovime seda teha nädalavahetustel. Samuti tuleb laadimise ajaks välja lülitada andmete regulaarlaadimise käivitamine. Vt [Automaatprotsesside käivitamine](#)

Täislaadimiste teostamiseks tabelile TABEL1 tuleb andmelaos andmebaasos käivitada järgmine käsk (käsus muuta tabeli nimi ja vajaduse korral ka skeemi nimi; soovi korral võib korraga muuta mitme erineva tabeli laadimist)

```
update adm.tables t1
  set
    load_method = 'REWRITE',
    modified_filter = null,
    last_load_expression = null
where source_table_name in ('TABEL1')
   and source_schema_name = 'SKA_OWNER'
   and is_valid = 1
```

Pärast täislaadimise lõppemist tuleb taas sisse lülitada regulaarlaadimine, Tabeli TABEL1 regulaarlaadimise sisselülitamiseks tuleb käivitada järgmine käsk (käsus muuta tabeli nimi ja vajaduse korral ka skeemi nimi):

```
update adm.tables t1
  set
    load_method = t2.load_method,
    modified_filter = t2.modified_filter,
    last_load_expression = t2.last_load_expression
from adm_backup.tables t2
where t1.is_valid = 1
   and t1.data_source_code = t2.data_source_code and t1.source_schema_name = t2.source_schema_name and t1.
source_table_name = t2.source_table_name and t2.is_valid = 1
   and t1.source_schema_name = 'SKA_OWNER'
   and t1.source_table_name in ('TABEL1')
```

Logimine

Vertica laadimiste logid

Vertica sql (wrh-db.prd.tehik.ee)

[ADM_SKAIS1.LOAD_LOG](#) ja [ADM_SKAIS2.LOAD_LOG](#) - laadimiste logitabelid. Vt täpsem kirjeldus logitabelite kirjelduse juurest.

[ADM_SKAIS1.EVENT_LOG](#) ja [ADM_SKAIS1.EVENT_LOG](#) - sündmuste logi (võimaldab leida täpsema Pentaho protseduuri ja sõlme, kus viga tekkis).

```
select * from adm_skais1.load_log where status != 'SUCCESS' order by log_id desc;
```

```
select * from adm_skais1.event_log order by event_id desc;
```

LOAD_LOG näide

ERROR ja NOT_LOADED on staatused, mis tähendavad päris viga.

LOG_ID	PROCESS_ID	SUBPROCESS_CODE	DATA_SOURCE_CODE	RELATED_OBJECT	START_DT	END_DT	STATUS	IS_IGNORE
398,679	997	ODS_LOAD	SKAIS1	SKA_OWNER.YLEMAKSED	2025-11-06 04:00:22.934	2025-11-06 04:00:26.988	SUCCESS	0

EVENT_LOG näide

EVENT_ID	EVENT_DT	EVENT_TYPE	EVENT_CLASS	EVENT_CODE	EVENT_MESSAGE	RELATED_OBJECT	PROCESS_NAME	STEP
_NAME	PROCESS_ID	DATA_SOURCE_CODE						
844	2025-05-30 11:39:41.967	ERROR	ODS_LOAD	ERROR_ODS_LOAD	ERROR in loading ODS table	SKA_OWNER.		
PENSIONARID	load_one_table_from_db_ODS_LOAD	prepare_and_run_ods_load	838	SKAIS1				

Serveri pentaho logid

Serveris pentaho-vertica-1a logi failid asuvad /var/logs/pentaho. Logifail on kuupäeva eesliitega kujul DDMMYY-load_all_skais1_live.log

/var/logs/pentaho/170422-load_all_skais1_live.log

Tasub otsida märksõnu ERROR, Abort, E=1

SKAIS2 andmeladu

Laadimiste haldus

Tabelite laadimisse lisamine

Antud juhend kehtib ka Skais1 puhul, kuid skeemideks sel juhul ADM_SKAIS1

1. Tabelid lisatakse laadimisse osalt laadimismootori poolelt automaatselt. Laadimiste kasutajal on ligipääs skeemile, tabelile ning laadimismootor loeb tabeli metaadandmed (*metadata*).

Laadimismootor täidab automaatselt skeemi ADM_SKAIS2 tabelid COLUMNS, SCHEMAS, TABLES metaandmetega

2. Kontrollida, et vastav tabel on tabelis TABLES esindatud ning muuta selle LOAD_STATUS, LOAD_METHOD ja vajaduse korral muud laadimis parameetrid sobivaks.

(Väljade MODIFIED_FILTER ja LAST_LOAD_EXPRESSION väärtused sõltuvad sellest, millised väljad on antud tabelis nt millise tingimusega tuleb pärida muutunud andmed jms).

Vt täpsemalt [TABLES SKAIS2](#)

```
update ADM_SKAIS2.TABLES
set LOAD_STATUS = 'LOAD',
  LOAD_METHOD = 'LOAD_MODIFIED',
  MODIFIED_FILTER = '(case when MUUTM_AEG < DATE "1900-01-01" then DATE "1900-01-01" else MUUTM_AEG end)
>=TO_DATE("[LAST_LOAD_DT]", "yyyy-mm-dd hh24:mi:ss")',
  LAST_LOAD_EXPRESSION = 'timestampadd("hour",-1, max(MUUTM_AEG))',
  IS_ROW_COUNT = 1,
  DWH_SCHEMA_NAME = 'DW_SKAIS2'
where is_valid = 1
and SOURCE_SCHEMA_NAME = 'SKAIS2'
and SOURCE_TABLE_NAME = 'IS_KONTAKT_STAATUS';
```

3. Lisaks põhitabelile tuleb laadimisse lülitada vastav LOG tabel

```
update ADM_SKAIS2.TABLES
set LOAD_STATUS = 'LOAD',
  LOAD_METHOD = 'LOAD_MODIFIED_WOID',
  MODIFIED_FILTER = 'LOG_ACTION in ("D","I") and LOG_LISAM_AEG >=TO_DATE("[LAST_LOAD_DT]", "yyyy-mm-dd hh24:
mi:ss")',
  LAST_LOAD_EXPRESSION = 'timestampadd("hour",-1, max(LOG_LISAM_AEG))',
  IS_ROW_COUNT = 0,
  DWH_SCHEMA_NAME = 'DW_SKAIS2',
  PRIORITY_STG = 110,
  PRIORITY_ODS = 90
where is_valid = 1
and SOURCE_SCHEMA_NAME = 'SKAIS2'
and SOURCE_TABLE_NAME = 'LOG_IS_KONTAKT_STAATUS';
```

4. Lülitada sisse vastavate tabelite veergude laadimine (tabelis ADM_SKAIS2.COLUMNS) nii tava kui LOG tabeli puhul

```
update ADM_SKAIS2.COLUMNS
set LOAD_STATUS = 'LOAD'
where is_valid = 1
and SOURCE_SCHEMA_NAME = 'SKAIS2'
and SOURCE_TABLE_NAME in ('IS_KONTAKT_STAATUS', 'LOG_IS_KONTAKT_STAATUS');
```

Uus skeem

Uue skeemi lisandumisel tellida see itabi kaudu vastavalt:

[Ligipääsud](#)

Vead ja parandused

BLOB väljade puhul liiga pikad väärtused. Pikendada välja või muuta pikemaks tüübiks

```

--rows were rejected. Blob. varchar(64000) to long varchar
ALTER TABLE DW_SKAIS_PARENTAL_BENEFIT_STG.STG_SOURCE_INFORMATION add column DATA2 LONG VARCHAR
(300000);
UPDATE DW_SKAIS_PARENTAL_BENEFIT_STG.STG_SOURCE_INFORMATION SET DATA2 = DATA;
ALTER TABLE DW_SKAIS_PARENTAL_BENEFIT_STG.STG_SOURCE_INFORMATION RENAME COLUMN DATA TO
DATA_BAK;
ALTER TABLE DW_SKAIS_PARENTAL_BENEFIT_STG.STG_SOURCE_INFORMATION RENAME COLUMN DATA2 TO DATA;
select count(*) from DW_SKAIS_PARENTAL_BENEFIT_ODS.SOURCE_INFORMATION;
ALTER TABLE DW_SKAIS_PARENTAL_BENEFIT_STG.STG_SOURCE_INFORMATION DROP COLUMN DATA_BAK;

```

RAW andmetüübi puhul on vaja lisada transform expression

```

select * from ADM_SKAIS2.COLUMNS
where is_valid = 1
and LOAD_STATUS in ('LOAD', 'NEW_LOAD')
and data_type = 'RAW'
and TRANSFORM_EXPRESSION is NULL
and SOURCE_TABLE_NAME in ('IS_KONTAKT_STAATUS', 'LOG_IS_KONTAKT_STAATUS');

update ADM_SKAIS2.COLUMNS
set TRANSFORM_EXPRESSION = 'rawtohex(' || dwh_column_name || ')'
where is_valid = 1
and LOAD_STATUS in ('LOAD', 'NEW_LOAD')
and data_type = 'RAW'
and TRANSFORM_EXPRESSION is NULL;

```

Timestamp with timezone puhul vajalik transform expression

```

update ADM_SKAIS2.COLUMNS
set TRANSFORM_EXPRESSION = 'cast(cast(' || dwh_column_name || ' as timestamp with local time zone) as timestamp)'
where is_valid = 1
and LOAD_STATUS = 'LOAD'
and data_type = 'TIMESTAMP(3) WITH TIME ZONE'
and TRANSFORM_EXPRESSION is NULL
and SOURCE_TABLE_NAME in ('IS_KONTAKT_STAATUS', 'LOG_IS_KONTAKT_STAATUS');

```

Logid

Vertica laadimiste logid

Vertica sql (wrh-db.prd.tehik.ee)

ADM_SKAIS2.EVENT_LOG

ADM_SKAIS2.LOAD_LOG

```
select * from adm_skais2.load_log where status != 'SUCCESS' order by log_id desc;
```

```
select * from adm_skais2.event_log order by event_id desc;
```

LOAD_LOG näide

Warning staatus on okei, sest see enamasti tähendab kirjete arvu erinevust. ERROR ja NOT LOADED on päris vead.

```
LOG_ID PROCESS_ID SUBPROCESS_CODE DATA_SOURCE_CODE RELATED_OBJECT START_DT END_DT STATUS IS_IGNORE  
896,315 879 ODS_LOAD SKAIS2 SKAIS2.YL_YLESANNE_SEOS 2025-10-30 04:16:29.909 2025-10-30 04:16:39.947 WARNING 0
```

EVENT_LOG näide

```
EVENT_ID EVENT_DT EVENT_TYPE EVENT_CLASS EVENT_CODE EVENT_MESSAGE RELATED_OBJECT PROCESS_NAME STEP  
_NAME PROCESS_ID DATA_SOURCE_CODE  
3,991 2025-09-16 03:11:52.732 ERROR ODS_LOAD ERROR_ODS_LOAD_STG_ERROR STG table loading error - ODS table not  
loaded SKAIS2.AE_HINDAMINE_VALDKOND load_one_table_from_db_ODS_LOAD Is STG table OK 844 SKAIS2
```

Serveri pentaho logid

Serveris pentaho-vertica-1a logi failid asuvad /var/logs/pentaho. Logifail on kuupäeva eesliitega kujul DDMMYY-load_all_skais2_live.log

/var/logs/pentaho/170422-load_all_skais2_live.log

Logidest tasub otsida märksõna ERROR, Abort, E=1

Pentaho PDI

Pentaho PDI tarkvara

Pentaho PDI 9.4 versioon pole enam allalaetav tootja poolt. Edasised versioonid on tasuta. Tarkvara leiab serveritest ja Resta arhiividest.

Asukohad serverites

SKAIS- pentaho-vertica-1a
/opt/etl/data-integration94

Paigaldus

1. Laadida pdi-ce-9.4.0.0-343.zip serverisse ning lahti pakkida soovitud asukohta. PDI ise võtab ~2GB ruumi ning sisemised logid võivad veel mitu GB võtta.
2. Sättida õigused "pentaho" kasutajal
chown -R pentaho /opt/etl/data-integration94
chmod u+x /opt/etl/data-integration94/*.sh
3. Paigaldada jdbc andmebaasi draiverid lib kausta
Näiteks:
/opt/etl/data-integration94/lib/ojdbc8.jar
/opt/etl/data-integration94/lib/vertica-jdbc-12.0.3-0.jar
4. Sättida mälu piirid vastavalt serveri võimekustele.
NB PDI võib tekitada mitu protsessi ning mitmed laadimised võivad samaaegselt käia.
vi /opt/etl/data-integration94/spoon.sh
if [-z "\$PENTAHO_DI_JAVA_OPTIONS"]; then
PENTAHO_DI_JAVA_OPTIONS="-Xms1024m -Xmx16624m"

Vajadusel määrata repositooriumid
vi /home/pentaho/.kettle/repositories.xml

Näiteks:

```
<repository>  
  <id>KettleFileRepository</id>  
  <name>skais2_live</name>  
  <description>SKAIS2 live repo</description>  
  <is_default>false</is_default>  
  <base_directory>/home/pentaho/repo_skais2_v2_live</base_directory>  
  <read_only>N</read_only>  
  <hides_hidden_files>N</hides_hidden_files>  
</repository>
```

Vajadusel kirjeldada muutujad kettle.properties failis kujul MUUTUJANIMI=vaartus
/home/pentaho/kettle/kettle.properties

Kui laadimiste osa on ssh/sftp vms pöördumised teise serverisse, siis vaja konfigureerida ssh võtmega ligipääs.

SKA SKAIS2 DW andmebaas

SKA DW andmebaasi (SKA andmelao) andmebaasisüsteemina kasutatakse spetsiaalselt analüütilisteks lahendusteks optimeeritud veerupõhist andmebaasimootorit OpenText Vertica.

SKA DW andmebaas jaguneb andmete laadimise loogika mõttes kahte kihti:

1. Eellaadimise ala (staging ehk STG-kiht) sisaldab andmete laadimise tööks vajalikke andmetabeleid (ajutised andmed). STG-kihi andmetabelite struktuur on täpselt sama, mis on lähtesüsteemi andmetabelitel (mõningased erinevused võivad tulla sellest, kuidas erinevad andmebaasisüsteemid (Oracle ja Vertica) salvestavad sarnaseid andmetüüpe nt varchar2(100) Oracles vs varchar(100) Verticas). Tüüpiliselt laaditakse igaõise laadimise käigus STG-kihti viimase ööpäeva jooksul lähtesüsteemidesse lisatud ja/või seal muudetud andmed. Erandjuhtumitel (kui lähtesüsteemis muudetud kirjade tuvastamine ei ole võimalik), teostatakse tabelile igal öösel täislaadimine. Eeskirjad, millist andmete laadimise meetodit ja see, kas vastava tabeli andmeid üldse SKAIS2 andmelattu laaditakse, on kirjeldatud süsteemses andmetabelis ADM_SKAIS2.TABLES.
2. Operatiivne andmehoidla (Operational Data Store e ODS-kiht) sisaldab kõiki lähtesüsteemide andmebaaside andmeid (st kogu muudatuste ajalugu alates andmelao laadimise alghetkest). ODS kihi andmemudel on sarnane lähtesüsteemi andmemudelitele - igale lähtesüsteemi tabelile vastab ODS kihis samanimeline tabel, mis sisaldab kõiki vastava andmetabeli välju ning nelja täiendavat välja kirjade versioneerimiseks. Lisatavateks väljadeks on:
 - a. dwh_id - Surrogaatvõti (andmelao laadimisprogrammi poolt genereeritud võtmeväli)
 - b. is_valid - 0/1 väli, mis näitab, kas tegemist on kehtiva kirjega (kõige viimase versiooniga)
 - c. valid_from - versiooni kehtivuse algusaeg (kuupäev ja kellaeg)
 - d. valid_to - versiooni kehtivuse lõppaeg (kuupäev ja kellaeg)

Lisaks nimetatud kihtidele on eraldi skeem (nimega ADM_SKAIS2), mis sisaldab andmete laadimiseks ning muuks SKA DW toimimiseks vajalikke andmetabeleid.

SKA SKAIS2 DW andmete laadimine

Andmete laadimiste üldine loogika

SKA_DW andmete laadimine toimub järgmise skeemi kohaselt

1. Lähtesüsteemi (SKAIS2) andmemudeli muudatuste tuvastamine (sh SKA DW andmemudeli uuendamine)
 - a. Lähteallika (SKAIS2) andmebaasi süsteemsetest tabelitest loetakse lähtesüsteemi andmemudel ning salvestatakse see süsteemsetesse tabelitesse ADM_SKAIS2.IMP_SCHEMAS, ADM_SKAIS2.IMP_TABLES ja ADM_SKAIS2.IMP_COLUMNS.
 - b. Lähteallika andmemudelit võrreldakse SKA DW poolt teada oleva mudeliga (st eelmisel päeval kehtinud andmemudeliga) ning tuvastatakse muudatused.
 - c. Kõikide nende andmetabelite korral, mis on laadimisse sisse lülitatud (LOAD_STATUS = 'LOAD' või 'NEW_LOAD') korral SKA DW andmemudelit (STG-tabeleid ja ODS-tabeleid) muudetakse vastavalt eelmises punktis tuvastatud muudatustele (lisatakse uued väljad, muudetakse olemasolevate väljade tüüpe, lähtesüsteemist kustutatud väljade laadimine lülitatakse välja (LOAD_STATUS = 'NO_OBJECT')).
2. Andmete import ja andmete laadimine STG ja ODS tabelitesse (iga tabeli korral võidakse kasutada erinevat andmete impordi meetodit, erinevat lähtesüsteemist kustutatud kirjade tühistatuse märkimise meetodit jms. Vt tabelite laadimise konfigureerimise võimalusi süsteemsete tabelite [ADM_SKAIS2.TABLES](#) ja [ADM_SKAIS2.COLUMNS](#) kirjelduste juurest)
 - a. (Pärast viimast laadimist lisatud ja/või muudetud) andmete import STG tabelitesse (sh viimati laaditud andmete tuvastamine kasutades ADM_SKAIS2.TABLES LAST_LOAD_EXPRESSION avaldist)
 - b. Andmete laadimine ODS tabelitesse (sh lähtesüsteemist kustutatud kirjade märkimine kehtetuks)
 - c. Lähtesüsteemi ja ODS tabeli kirjade arvu võrdlemine.

Andmelao laadimiste kohta tekib laadimiste logi. Laadimisprotsesside käivitamise ja lõpetamise teated salvestatakse laadimise logi tabelisse [LOAD_LOG](#), detailsemad tabelite loomise (sündmuste teated) salvestatakse tabelisse [EVENT_LOG](#).

Laadimiste käitumine veaolukordades

Laadimise alguses kontrollitakse, kas eelmine (sama andmeallika) laadimine on lõppenud. Kui eelmine laadimise protsess ei ole lõppenud või lõppes veaga ning ei ole märgitud, et viga tuleb ignoreerida, siis järgmist sama allika laadimisprotsessi ei käivitata.

Kui eelmise päeva laadimine sai vea ning vigane olukord on parandatud (st järgmisel päeval saab laadimist uuesti jätkata), siis tuleb laadimiste logi tabelis (LOAD_LOG) IS_IGNORE väärtuseks panna 1 (st viga ignoreeritakse).

Juhul, kui mingi STG tabeli laadimise käigus tekib viga, siis STG tabelite laadimine töötab lõpuni, samuti käivitatakse ODS tabelite laadimine kuid vea saanud tabeli korral vastava ODS tabeli laadimist ei käivitata.

Laadimiste tulemustest teavitamine

Laadimiste tulemuste teavitamiseks saadetakse e-mail aadressile (vastava e-mail'i aliase loob ning selle aliase taga paiknevaid konkreetseid e-mail'i aadresse haldab TEHIK, e-mail'i alias tuleb sisestada tabelisse PARAMETERS parameetri to_email väärtuseks).

Juhul, kui laadimiste käigus ei tekkinud vigu ega hoiatusi, saadetakse e-mail teemaga "Andmeallika SKAIS2 andmete laadimine õnnestus".

Juhul, kui laadimise käigus tekkis hoiatusi, saadetakse e-mail teemaga "Andmeallika SKAIS2 andmete laadimine on lõppenud hoiatustega".

Juhul, kui laadimise käigus tekkis vigu, saadetakse e-mail teemaga "Andmeallika SKAIS2 laadimine sai vigu!" ning e-mail'i sisuks on vigade logi.

Juhul, kui tavapäraseks laadimise lõppemise ajaks ei ole e-mail'i saabunud, siis tuleb olukorda käsitleda veaolukorrana ning täpsemat infot on võimalik leida logitabelitest ja/või Pentaho logist.

Laaditavad tabelid

Kõige täpsema ülevaata, milliseid andmetabeleid (ja milliste meetoditega) laaditakse, saab süsteemsest andmetabelist ADM_SKAIS2.TABLES alljärgneva päringuga

```
select source_schema_name || '.' || source_table_name,
       dwh_schema_name || '.' || dwh_table_name,
       load_method, modified_filter
from adm_skais2.tables
where is_valid = 1
      and load_status in ('LOAD', 'NEW_LOAD')
order by source_schema_name, source_table_name
```

Alljärgnevas tabelis on laaditavate tabelite nimekiri 30.10.2025 seisuga

SOURCE_TABLE	DWH_TABLE	LOAD_METHOD	MODIFIED_FILTER
SKAIS2. AD_AADRESS	DW_SKAIS2. AD_AADRESS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_ARST_SPETSIA LIST	DW_SKAIS2. AE_ARST_SPETSIA LIST	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_ARST_SPETSIA LIST_DOKUMENT	DW_SKAIS2. AE_ARST_SPETSIA LIST_DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_EKSPERTARST LEPING	DW_SKAIS2. AE_EKSPERTARST LEPING	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2.AE_ERIALA	DW_SKAIS2. AE_ERIALA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINDAMINE_TE GEVUS	DW_SKAIS2. AE_HINDAMINE_TE GEVUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINDAMINE_TE GEVUS_KYSIMUS	DW_SKAIS2. AE_HINDAMINE_TE GEVUS_KYSIMUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINDAMINE_VA LDKOND	DW_SKAIS2. AE_HINDAMINE_VA LDKOND	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINDAMINE_VA LDKOND_TASE	DW_SKAIS2. AE_HINDAMINE_VA LDKOND_TASE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG	DW_SKAIS2. AE_HINNANG	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_DIA GNOOS	DW_SKAIS2. AE_HINNANG_DIAG NOOS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_MOJ UTATUD	DW_SKAIS2. AE_HINNANG_MOJ UTATUD	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_OLEK	DW_SKAIS2. AE_HINNANG_OLEK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_TEG EVUS	DW_SKAIS2. AE_HINNANG_TEG EVUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_TEG EVUS_VASTUS	DW_SKAIS2. AE_HINNANG_TEG EVUS_VASTUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_VAL DKOND	DW_SKAIS2. AE_HINNANG_VAL DKOND	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_HINNANG_YTL US	DW_SKAIS2. AE_HINNANG_YTL US	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_LEPING_PUUD UMINE	DW_SKAIS2. AE_LEPING_PUUD UMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_LEPING_TOO	DW_SKAIS2. AE_LEPING_TOO	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_PIKK_TEKST	DW_SKAIS2. AE_PIKK_TEKST	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_TEHTUD_TOO	DW_SKAIS2. AE_TEHTUD_TOO	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_TOO_LIIK	DW_SKAIS2. AE_TOO_LIIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AE_TOO_LIIK_HIND	DW_SKAIS2. AE_TOO_LIIK_HIND	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')

SKAIS2. AK_AKTSEPTEERI MINE	DW_SKAIS2. AK_AKTSEPTEERI MINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2.AR_ARVE	DW_SKAIS2. AR_ARVE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_ARVESTUS_MU UDATUS	DW_SKAIS2. AR_ARVESTUS_MU UDATUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_ARVE_RIDA	DW_SKAIS2. AR_ARVE_RIDA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_FINANTS_SYN DMUS	DW_SKAIS2. AR_FINANTS_SYND MUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KATUSNOUE	DW_SKAIS2. AR_KATUSNOUE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPEETUD_ SUMMA	DW_SKAIS2. AR_KINNIPEETUD_ SUMMA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMINE	DW_SKAIS2. AR_KINNIPIDAMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_DOKUMENT	DW_SKAIS2. AR_KINNIPIDAMINE _DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_HYVITIS	DW_SKAIS2. AR_KINNIPIDAMINE _HYVITIS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_HYV_REEGEL	DW_SKAIS2. AR_KINNIPIDAMINE _HYV_REEGEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_ISIK	DW_SKAIS2. AR_KINNIPIDAMINE _ISIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_OLEK	DW_SKAIS2. AR_KINNIPIDAMINE _OLEK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_REEGEL	DW_SKAIS2. AR_KINNIPIDAMINE _REEGEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_REEGEL_DOK	DW_SKAIS2. AR_KINNIPIDAMINE _REEGEL_DOK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KINNIPIDAMIN E_REEGEL_HYV	DW_SKAIS2. AR_KINNIPIDAMINE _REEGEL_HYV	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KOHUSTUS	DW_SKAIS2. AR_KOHUSTUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KOHUSTUS_M UUDATUS	DW_SKAIS2. AR_KOHUSTUS_M UUDATUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KOHUSTUS_TY HISTUS_VIGA	DW_SKAIS2. AR_KOHUSTUS_TY HISTUS_VIGA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KOHU_NOUE_ SYNDMUS	DW_SKAIS2. AR_KOHU_NOUE_S YNDMUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_KOHU_VALJAM AKSE	DW_SKAIS2. AR_KOHU_VALJAM AKSE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_MAKSEGRAAFIK	DW_SKAIS2. AR_MAKSEGRAAFIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_MAKSEGRAAFI K_DOKUMENT	DW_SKAIS2. AR_MAKSEGRAAFI K_DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. AR_MAKSEGRAAFI K_RIDA	DW_SKAIS2. AR_MAKSEGRAAFI K_RIDA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	

SKAIS2. AR_MAKSEGRAAFI K_RIDA_LAEKUMINE	DW_SKAIS2. AR_MAKSEGRAAFI K_RIDA_LAEKUMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MAKSEKANAL	DW_SKAIS2. AR_MAKSEKANAL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MAKSEKANAL_ REEGEL	DW_SKAIS2. AR_MAKSEKANAL_ REEGEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MAKSM_KOHU STUS	DW_SKAIS2. AR_MAKSM_KOHU STUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MAKSULEPING	DW_SKAIS2. AR_MAKSULEPING	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MITTERESIDENT	DW_SKAIS2. AR_MITTERESIDENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MITTERES_MA KSULEPING_TOEND	DW_SKAIS2. AR_MITTERES_MA KSULEPING_TOEND	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MVT_PARING_I SIK	DW_SKAIS2. AR_MVT_PARING_I SIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MVT_SALDO	DW_SKAIS2. AR_MVT_SALDO	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_MVT_SYNDMUS	DW_SKAIS2. AR_MVT_SYNDMUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2.AR_NOUE	DW_SKAIS2. AR_NOUE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_NOUE_LAEKU MINE	DW_SKAIS2. AR_NOUE_LAEKU MINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_PANGAKONTO	DW_SKAIS2. AR_PANGAKONTO	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2.AR_PANK	DW_SKAIS2. AR_PANK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_PANK_PYHA	DW_SKAIS2. AR_PANK_PYHA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_PARAMEETER_ HYVITIS	DW_SKAIS2. AR_PARAMEETER_ HYVITIS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_PROTSESS_VE ATEADE	DW_SKAIS2. AR_PROTSESS_VE ATEADE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_RAHALINE_TE HING	DW_SKAIS2. AR_RAHALINE_TE HING	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_REGRESSINOUE	DW_SKAIS2. AR_REGRESSINOUE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_SAP_HANKIJA	DW_SKAIS2. AR_SAP_HANKIJA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_TAGASINOUE	DW_SKAIS2. AR_TAGASINOUE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_TAGASINOUE_ DOKUMENT	DW_SKAIS2. AR_TAGASINOUE_ DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_TAGASINOUE_ HYVITIS	DW_SKAIS2. AR_TAGASINOUE_ HYVITIS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_TAGASINOUE_I SIK	DW_SKAIS2. AR_TAGASINOUE_I SIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_TSD_ESD_KP	DW_SKAIS2. AR_TSD_ESD_KP	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. AR_TSD_ESD_SM	DW_SKAIS2. AR_TSD_ESD_SM	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')

SKAIS_DOCUMENT _MANAGEMENT. DOCUMENT	DW_SKAIS_DOCUM ENT_MANAGEMEN T.DOCUMENT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. DOCUMENT_BASIS	DW_SKAIS_DOCUM ENT_MANAGEMEN T. DOCUMENT_BASIS	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. DOCUMENT_DOMA IN	DW_SKAIS_DOCUM ENT_MANAGEMEN T. DOCUMENT_DOMA IN	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. DOCUMENT_FILE	DW_SKAIS_DOCUM ENT_MANAGEMEN T.DOCUMENT_FILE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. DOCUMENT_PERS ON	DW_SKAIS_DOCUM ENT_MANAGEMEN T. DOCUMENT_PERS ON	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. LOG_DOCUMENT	DW_SKAIS_DOCUM ENT_MANAGEMEN T.LOG_DOCUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy- mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. LOG_DOCUMENT_ BASIS	DW_SKAIS_DOCUM ENT_MANAGEMEN T. LOG_DOCUMENT_ BASIS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy- mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. LOG_DOCUMENT_ DOMAIN	DW_SKAIS_DOCUM ENT_MANAGEMEN T. LOG_DOCUMENT_ DOMAIN	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy- mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. LOG_DOCUMENT_ FILE	DW_SKAIS_DOCUM ENT_MANAGEMEN T. LOG_DOCUMENT_ FILE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy- mm-dd hh24:mi:ss')
SKAIS_DOCUMENT _MANAGEMENT. LOG_DOCUMENT_ PERSON	DW_SKAIS_DOCUM ENT_MANAGEMEN T. LOG_DOCUMENT_ PERSON	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy- mm-dd hh24:mi:ss')
SKAIS2. ERI_TOOTASU_HY VITAMINE	DW_SKAIS2. ERI_TOOTASU_HY VITAMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. ERI_TOOTASU_HY VI_LAPS	DW_SKAIS2. ERI_TOOTASU_HY VI_LAPS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. ERP_S_RAHALINE_ TEHING	DW_SKAIS2. ERP_S_RAHALINE_ TEHING	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. ERP_S_RAHALINE_ TEHING_VIGA	DW_SKAIS2. ERP_S_RAHALINE_ TEHING_VIGA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. ERP_V_KOHUSTUS _SYNDMUS	DW_SKAIS2. ERP_V_KOHUSTUS _SYNDMUS	LOAD_MODIFIED	(case when to_timestamp(MUUTM_AEG) < DATE '1900-01-01' then DATE '1900-01-01' else to_timestamp(MUUTM_AEG) end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. ERP_V_NOUE_SYN DMUS	DW_SKAIS2. ERP_V_NOUE_SYN DMUS	LOAD_MODIFIED	(case when to_timestamp(MUUTM_AEG) < DATE '1900-01-01' then DATE '1900-01-01' else to_timestamp(MUUTM_AEG) end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. HY_HYVITIS	DW_SKAIS2. HY_HYVITIS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. HY_HYVITIS_DOKU MENT	DW_SKAIS2. HY_HYVITIS_DOKU MENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. HY_HYVITIS_ISIK	DW_SKAIS2. HY_HYVITIS_ISIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. HY_HYVITIS_ISIK_ KITSENDUS	DW_SKAIS2. HY_HYVITIS_ISIK_ KITSENDUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')

SKAIS2. IS_POORDUMINE	DW_SKAIS2. IS_POORDUMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_POORDUMINE_ DOKUMENT	DW_SKAIS2. IS_POORDUMINE_ DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_PUUDUMINE	DW_SKAIS2. IS_PUUDUMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_STAATUS	DW_SKAIS2. IS_STAATUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_STAATUS_DOK UMENT	DW_SKAIS2. IS_STAATUS_DOKU MENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_TEAVITUS	DW_SKAIS2. IS_TEAVITUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_TEAVITUS_OLEK	DW_SKAIS2. IS_TEAVITUS_OLEK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_TOOVOIME	DW_SKAIS2. IS_TOOVOIME	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_TULU_ALUS	DW_SKAIS2. IS_TULU_ALUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_TULU_DOKUME NT	DW_SKAIS2. IS_TULU_DOKUME NT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_VOLITUS	DW_SKAIS2. IS_VOLITUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. IS_VOLITUS_DOKU MENT	DW_SKAIS2. IS_VOLITUS_DOKU MENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KA_KASUTAJA_RO LL_PRIVILEEG	DW_SKAIS2. KA_KASUTAJA_RO LL_PRIVILEEG	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2.KA_SEOS	DW_SKAIS2. KA_SEOS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KA_TOKEN_LUBA	DW_SKAIS2. KA_TOKEN_LUBA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KA_TOKEN_SAADE TUD	DW_SKAIS2. KA_TOKEN_SAADE TUD	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KL_ELEMENT	DW_SKAIS2. KL_ELEMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KL_ELEMENT_SEOS	DW_SKAIS2. KL_ELEMENT_SEOS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KL_ELEMENT_TEK ST	DW_SKAIS2. KL_ELEMENT_TEK ST	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. KL_KLASSIFIKAAT OR	DW_SKAIS2. KL_KLASSIFIKAAT OR	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AD_AADDRESS	DW_SKAIS2. LOG_AD_AADDRESS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AE_ARST_SP ETSIALIST	DW_SKAIS2. LOG_AE_ARST_SP ETSIALIST	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AE_ARST_SP ETSIALIST_DOKUME	DW_SKAIS2. LOG_AE_ARST_SP ETSIALIST_DOKUME	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AE_EKSPERT ARST_LEPING	DW_SKAIS2. LOG_AE_EKSPERT ARST_LEPING	LOAD_MODIFIED_ WOID	(DOKUMENT_ID,LOG_LISAM_AEG) in (select DOKUMENT_ID, max(LOG_LISAM_AEG) from SKAIS2.LOG_AE_EKSPERTARST_LEPING where DOKUMENT_ID in (select DOKUMENT_ID from SKAIS2.LOG_AE_EKSPERTARST_LEPING where log_action='D') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss') group by DOKUMENT_ID)	
SKAIS2. LOG_AE_ERIALA	DW_SKAIS2. LOG_AE_ERIALA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AE_HINDAMI NE_TEGEVUS	DW_SKAIS2. LOG_AE_HINDAMI NE_TEGEVUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	

SKAIS2. LOG_AE_HINDAMI NE_TEGEVUS_KYS IM	DW_SKAIS2. LOG_AE_HINDAMIN E_TEGEVUS_KYSIM	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINDAMI NE_VALDKOND	DW_SKAIS2. LOG_AE_HINDAMIN E_VALDKOND	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINDAMI NE_VALDKOND_TA SE	DW_SKAIS2. LOG_AE_HINDAMIN E_VALDKOND_TA SE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG	DW_SKAIS2. LOG_AE_HINNANG	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _DIAGNOOS	DW_SKAIS2. LOG_AE_HINNANG _DIAGNOOS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _MOJUTATUD	DW_SKAIS2. LOG_AE_HINNANG _MOJUTATUD	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _OLEK	DW_SKAIS2. LOG_AE_HINNANG _OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _TEGEVUS	DW_SKAIS2. LOG_AE_HINNANG _TEGEVUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _TEGEVUS_VASTUS	DW_SKAIS2. LOG_AE_HINNANG _TEGEVUS_VASTUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _VALDKOND	DW_SKAIS2. LOG_AE_HINNANG _VALDKOND	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_HINNANG _YTLUS	DW_SKAIS2. LOG_AE_HINNANG _YTLUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_LEPING_ PUUDUMINE	DW_SKAIS2. LOG_AE_LEPING_P UUDUMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_LEPING_ TOO	DW_SKAIS2. LOG_AE_LEPING_ TOO	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_PIKK_T EKST	DW_SKAIS2. LOG_AE_PIKK_T EKST	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_TEHTUD_ TOO	DW_SKAIS2. LOG_AE_TEHTUD_ TOO	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_TOO_LI IK	DW_SKAIS2. LOG_AE_TOO_LI IK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AE_TOO_LI IK_HIND	DW_SKAIS2. LOG_AE_TOO_LI IK_HIND	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AK_AKTSEPT EERIMINE	DW_SKAIS2. LOG_AK_AKTSEPT EERIMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AR_ARVE	DW_SKAIS2. LOG_AR_ARVE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AR_ARVESTU S_MUUDATUS	DW_SKAIS2. LOG_AR_ARVESTU S_MUUDATUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AR_ARVE_RI DA	DW_SKAIS2. LOG_AR_ARVE_RI DA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AR_FINANTS _SYNDMUS	DW_SKAIS2. LOG_AR_FINANTS_ SYNDMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_AR_KATUSN OUE	DW_SKAIS2. LOG_AR_KATUSNO UE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))

SKAIS2. LOG_AR_KINNIPEE TUD_SUMMA	DW_SKAIS2. LOG_AR_KINNIPEE TUD_SUMMA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE	DW_SKAIS2. LOG_AR_KINNIPID AMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_DOKUMENT	DW_SKAIS2. LOG_AR_KINNIPID AMINE_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_HYVITIS	DW_SKAIS2. LOG_AR_KINNIPID AMINE_HYVITIS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_HYV_REEGE	DW_SKAIS2. LOG_AR_KINNIPID AMINE_HYV_REEGE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_ISIK	DW_SKAIS2. LOG_AR_KINNIPID AMINE_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_OLEK	DW_SKAIS2. LOG_AR_KINNIPID AMINE_OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_REEGEL	DW_SKAIS2. LOG_AR_KINNIPID AMINE_REEGEL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_REEGEL_DO	DW_SKAIS2. LOG_AR_KINNIPID AMINE_REEGEL_DO	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KINNIPID AMINE_REEGEL_HY	DW_SKAIS2. LOG_AR_KINNIPID AMINE_REEGEL_HY	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KOHUST US	DW_SKAIS2. LOG_AR_KOHUSTUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KOHUST US_MUUDATUS	DW_SKAIS2. LOG_AR_KOHUSTU S_MUUDATUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KOHUST US_TYHISTUS_VIGA	DW_SKAIS2. LOG_AR_KOHUSTU S_TYHISTUS_VIGA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KOHU_N OUE_SYNDMUS	DW_SKAIS2. LOG_AR_KOHU_NO UE_SYNDMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_KOHU_VA LJAMAKSE	DW_SKAIS2. LOG_AR_KOHU_VA LJAMAKSE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSEG RAAFIK	DW_SKAIS2. LOG_AR_MAKSEG RAAFIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSEG RAAFIK_DOKUMENT	DW_SKAIS2. LOG_AR_MAKSEG RAAFIK_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSEG RAAFIK_RIDA	DW_SKAIS2. LOG_AR_MAKSEG RAAFIK_RIDA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSEG RAAFIK_RIDA_LAE KU	DW_SKAIS2. LOG_AR_MAKSEG RAAFIK_RIDA_LAE KU	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSEK ANAL	DW_SKAIS2. LOG_AR_MAKSEKA NAL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSEK ANAL_REEGEL	DW_SKAIS2. LOG_AR_MAKSEKA NAL_REEGEL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MAKSM_ KOHUSTUS	DW_SKAIS2. LOG_AR_MAKSM_K OHUSTUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')

SKAIS2. LOG_AR_MAKSULE PING	DW_SKAIS2. LOG_AR_MAKSULE PING	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MITTE RE SIDENT	DW_SKAIS2. LOG_AR_MITTE RE SIDENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MITTE RE S_MAKSULE PING_ TO	DW_SKAIS2. LOG_AR_MITTE RE S_MAKSULE PING_ TO	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MVT_P AR ING_ISIK	DW_SKAIS2. LOG_AR_MVT_P AR ING_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MVT_S AL DO	DW_SKAIS2. LOG_AR_MVT_S AL DO	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_MVT_S YN DMUS	DW_SKAIS2. LOG_AR_MVT_S YN DMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_NOUE	DW_SKAIS2. LOG_AR_NOUE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_NOUE_L A EKUMINE	DW_SKAIS2. LOG_AR_NOUE_L A EKUMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_PANGAK ONTO	DW_SKAIS2. LOG_AR_PANGAK ONTO	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_PANK	DW_SKAIS2. LOG_AR_PANK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_PANK_P Y HA	DW_SKAIS2. LOG_AR_PANK_P Y HA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_PARAM E ETER_HYVITIS	DW_SKAIS2. LOG_AR_PARAM E ETER_HYVITIS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_PROTSE S_ VEATEADE	DW_SKAIS2. LOG_AR_PROTSE S_ VEATEADE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_RAHAL IN E_TEHING	DW_SKAIS2. LOG_AR_RAHAL IN E_TEHING	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_REGRES SINOUE	DW_SKAIS2. LOG_AR_REGRES SINOUE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_SAP_H AN KIJA	DW_SKAIS2. LOG_AR_SAP_H AN KIJA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_SAP_V EA TEADE	DW_SKAIS2. LOG_AR_SAP_V EA TEADE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_TAGAS IN OUE	DW_SKAIS2. LOG_AR_TAGAS IN OUE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_TAGAS IN OUE_DOKUMENT	DW_SKAIS2. LOG_AR_TAGAS IN OUE_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_TAGAS IN OUE_HYVITIS	DW_SKAIS2. LOG_AR_TAGAS IN OUE_HYVITIS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_TAGAS IN OUE_ISIK	DW_SKAIS2. LOG_AR_TAGAS IN OUE_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_TSD_E SD _KP	DW_SKAIS2. LOG_AR_TSD_E SD _KP	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_AR_TSD_E SD _SM	DW_SKAIS2. LOG_AR_TSD_E SD _SM	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')

SKAIS2. LOG_AR_TSD_LISA _1A	DW_SKAIS2. LOG_AR_TSD_LISA _1A	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TSD_LISA _1B	DW_SKAIS2. LOG_AR_TSD_LISA _1B	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TSD_LISA _2A	DW_SKAIS2. LOG_AR_TSD_LISA _2A	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TSD_LISA _2B	DW_SKAIS2. LOG_AR_TSD_LISA _2B	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TSD_VORM	DW_SKAIS2. LOG_AR_TSD_VORM	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TULUMAKS	DW_SKAIS2. LOG_AR_TULUMAKS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TULUMAK SUVABASTUS	DW_SKAIS2. LOG_AR_TULUMAK SUVABASTUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_TULUMAK SUVAB_DOKUMENT	DW_SKAIS2. LOG_AR_TULUMAK SUVAB_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_VAHEMA KSE_PAEV	DW_SKAIS2. LOG_AR_VAHEMA KSE_PAEV	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_AR_VEATEAD E_OBJEKT	DW_SKAIS2. LOG_AR_VEATEAD E_OBJEKT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_DOKUME NT	DW_SKAIS2. LOG_DO_DOKUME NT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_DOKUME NT_FAIL	DW_SKAIS2. LOG_DO_DOKUME NT_FAIL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_DOKUME NT_OBJEKT	DW_SKAIS2. LOG_DO_DOKUME NT_OBJEKT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_DOKUME NT_SEOS	DW_SKAIS2. LOG_DO_DOKUME NT_SEOS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_DOKUME NT_SEOS_VALINE	DW_SKAIS2. LOG_DO_DOKUME NT_SEOS_VALINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_DOKUME NT_TEENUS	DW_SKAIS2. LOG_DO_DOKUME NT_TEENUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_KAART	DW_SKAIS2. LOG_DO_KAART	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_KYSIMUS TIK	DW_SKAIS2. LOG_DO_KYSIMUS TIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_KYSIMUS TIK_KYSIMUS	DW_SKAIS2. LOG_DO_KYSIMUS TIK_KYSIMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_KYSIMUS TIK_VASTUS	DW_SKAIS2. LOG_DO_KYSIMUS TIK_VASTUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_KYSIMUS TIK_VASTUS_VALIK	DW_SKAIS2. LOG_DO_KYSIMUS TIK_VASTUS_VALIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_OT SUS	DW_SKAIS2. LOG_DO_OT SUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_DO_OT SUS_A LUS	DW_SKAIS2. LOG_DO_OT SUS_A LUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')	

SKAIS2. LOG_DO_PIKK_TE KST	DW_SKAIS2. LOG_DO_PIKK_TEK ST	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_DO_TAOTLUS	DW_SKAIS2. LOG_DO_TAOTLUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_DO_TOOVOI METUSLEHT	DW_SKAIS2. LOG_DO_TOOVOIM ETUSLEHT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_ERI_TOOTAS U_HYVITAMINE	DW_SKAIS2. LOG_ERI_TOOTAS U_HYVITAMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_ERI_TOOTAS U_HYVI_LAPS	DW_SKAIS2. LOG_ERI_TOOTAS U_HYVI_LAPS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_ERP_S_RAHA LINE_TEHING	DW_SKAIS2. LOG_ERP_S_RAHA LINE_TEHING	LOAD_MODIFIED_ WOID	(ERP_TEHING_ID,LOG_LISAM_AEG) in (select ERP_TEHING_ID, max(LOG_LISAM_AEG) from SKAIS2.LOG_ERP_S_RAHALINE_TEHING where ERP_TEHING_ID in (select ERP_TEHING_ID from SKAIS2.LOG_ERP_S_RAHALINE_TEHING where log_action='D') and LOG_LISAM_AEG>=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss') group by ERP_TEHING_ID)
SKAIS2. LOG_ERP_S_RAHA LINE_TEHING_VIGA	DW_SKAIS2. LOG_ERP_S_RAHA LINE_TEHING_VIGA	LOAD_MODIFIED_ WOID	(ERP_VIGA_ID,LOG_LISAM_AEG) in (select ERP_VIGA_ID, max(LOG_LISAM_AEG) from SKAIS2.LOG_ERP_S_RAHALINE_TEHING_VIGA where ERP_VIGA_ID in (select ERP_VIGA_ID from SKAIS2.LOG_ERP_S_RAHALINE_TEHING_VIGA where log_action='D') and LOG_LISAM_AEG>=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss') group by ERP_VIGA_ID)
SKAIS2. LOG_ERP_V_KOHU STUS_SYNDMUS	DW_SKAIS2. LOG_ERP_V_KOHU STUS_SYNDMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_ERP_V_NOUE _SYNDMUS	DW_SKAIS2. LOG_ERP_V_NOUE _SYNDMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_HYVITIS	DW_SKAIS2. LOG_HY_HYVITIS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_HYVITIS_ DOKUMENT	DW_SKAIS2. LOG_HY_HYVITIS_ DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_HYVITIS_I SIK	DW_SKAIS2. LOG_HY_HYVITIS_I SIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_HYVITIS_I SIK_KITSENDUS	DW_SKAIS2. LOG_HY_HYVITIS_I SIK_KITSENDUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_HYVITIS_ RAHALINE	DW_SKAIS2. LOG_HY_HYVITIS_ RAHALINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_HYVITIS_ TEENUS	DW_SKAIS2. LOG_HY_HYVITIS_ TEENUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_MENETLUS	DW_SKAIS2. LOG_HY_MENETLUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_MENETLU S_DOKUMENT	DW_SKAIS2. LOG_HY_MENETLU S_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_MENETLU S_HYVITIS_OLEK	DW_SKAIS2. LOG_HY_MENETLU S_HYVITIS_OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_MENETLU S_ISIK	DW_SKAIS2. LOG_HY_MENETLU S_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_MENETLU S_SAMM	DW_SKAIS2. LOG_HY_MENETLU S_SAMM	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_HY_MENETLU S_SAMM_OLEK	DW_SKAIS2. LOG_HY_MENETLU S_SAMM_OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_IS_AMETNIK	DW_SKAIS2. LOG_IS_AMETNIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')

SKAIS2. LOG_IS_EEMALOL EK	DW_SKAIS2. LOG_IS_EEMALOL EK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK	DW_SKAIS2. LOG_IS_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_AADR ESS	DW_SKAIS2. LOG_IS_ISIK_AADR ESS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_DOK UMENT	DW_SKAIS2. LOG_IS_ISIK_DOKU MENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_KOD AKONDSUS	DW_SKAIS2. LOG_IS_ISIK_KODA KONDSUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_NOU SOLEK	DW_SKAIS2. LOG_IS_ISIK_NOUS OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_OLEK	DW_SKAIS2. LOG_IS_ISIK_OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_SEOS	DW_SKAIS2. LOG_IS_ISIK_SEOS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_SEOS _DOKUMENT	DW_SKAIS2. LOG_IS_ISIK_SEOS _DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_ISIK_TEOV OIME	DW_SKAIS2. LOG_IS_ISIK_TEOV OIME	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_KINNIPIDA MINE	DW_SKAIS2. LOG_IS_KINNIPIDA MINE	LOAD_MODIFIED_ WOID	(STAATUS_ID,LOG_LISAM_AEG) in (select STAATUS_ID, max(LOG_LISAM_AEG) from SKAIS2. LOG_IS_KINNIPIDAMINE where STAATUS_ID in (select STAATUS_ID from SKAIS2. LOG_IS_KINNIPIDAMINE where log_action='D') and LOG_LISAM_AEG>=TO_DATE (('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')) group by STAATUS_ID)
SKAIS2. LOG_IS_KOGUMIS PENSION	DW_SKAIS2. LOG_IS_KOGUMIS PENSION	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_KONTAKT	DW_SKAIS2. LOG_IS_KONTAKT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_MUU_IDEN TITEET	DW_SKAIS2. LOG_IS_MUU_IDEN TITEET	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_NOUSOLEK	DW_SKAIS2. LOG_IS_NOUSOLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_NOUSOLE K_DOKUMENT	DW_SKAIS2. LOG_IS_NOUSOLE K_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_OPPIMINE	DW_SKAIS2. LOG_IS_OPPIMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_OPPIMINE _OLEK	DW_SKAIS2. LOG_IS_OPPIMINE _OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_POORDUM INE	DW_SKAIS2. LOG_IS_POORDUM INE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_POORDUM INE_DOKUMENT	DW_SKAIS2. LOG_IS_POORDUM INE_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_PUUDUMI NE	DW_SKAIS2. LOG_IS_PUUDUMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_STAATUS	DW_SKAIS2. LOG_IS_STAATUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_STAATUS_ DOKUMENT	DW_SKAIS2. LOG_IS_STAATUS_ DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_TEAVITUS	DW_SKAIS2. LOG_IS_TEAVITUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))

SKAIS2. LOG_IS_TEAVITUS_OLEK	DW_SKAIS2. LOG_IS_TEAVITUS_OLEK	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_TOOVOIME	DW_SKAIS2. LOG_IS_TOOVOIME	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_TULU_ALUS	DW_SKAIS2. LOG_IS_TULU_ALUS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_TULU_DOKUMENT	DW_SKAIS2. LOG_IS_TULU_DOKUMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_VOLITUS	DW_SKAIS2. LOG_IS_VOLITUS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_IS_VOLITUS_DOKUMENT	DW_SKAIS2. LOG_IS_VOLITUS_DOKUMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KA_KASUTAJ_A_ROLL_PRIVILEEG	DW_SKAIS2. LOG_KA_KASUTAJ_A_ROLL_PRIVILEEG	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KA_SEOS	DW_SKAIS2. LOG_KA_SEOS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KA_TOKEN_LUBA	DW_SKAIS2. LOG_KA_TOKEN_LUBA	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KA_TOKEN_SAADETUD	DW_SKAIS2. LOG_KA_TOKEN_SAADETUD	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KL_ELEMENT	DW_SKAIS2. LOG_KL_ELEMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KL_ELEMENT_SEOS	DW_SKAIS2. LOG_KL_ELEMENT_SEOS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KL_ELEMENT_TEKST	DW_SKAIS2. LOG_KL_ELEMENT_TEKST	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_KL_KLASSIFIKAATOR	DW_SKAIS2. LOG_KL_KLASSIFIKAATOR	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_LX_LIIDES_VAARTUS_VASTAVUS	DW_SKAIS2. LOG_LX_LIIDES_VAARTUS_VASTAVUS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_MT_MALL	DW_SKAIS2. LOG_MT_MALL	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_MT_MALL_TEKST	DW_SKAIS2. LOG_MT_MALL_TEKST	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PA_PARAMETER	DW_SKAIS2. LOG_PA_PARAMETER	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PA_VAARTUS	DW_SKAIS2. LOG_PA_VAARTUS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PA_VAARTUS_M	DW_SKAIS2. LOG_PA_VAARTUS_M	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PE_MUU_PENSION	DW_SKAIS2. LOG_PE_MUU_PENSION	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PE_MUU_PENSION_DOKUMENT	DW_SKAIS2. LOG_PE_MUU_PENSION_DOKUMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PE_PENSIONILISA	DW_SKAIS2. LOG_PE_PENSIONILISA	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))
SKAIS2. LOG_PE_PENSIONILISA_DOKUMENT	DW_SKAIS2. LOG_PE_PENSIONILISA_DOKUMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss'))

SKAIS2. LOG_PE_PENSIONI LISA_ISIK	DW_SKAIS2. LOG_PE_PENSIONI LISA_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_SOODUS TUS	DW_SKAIS2. LOG_PE_SOODUST US	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_SOODUS TUS_DOKUMENT	DW_SKAIS2. LOG_PE_SOODUST US_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_SOODUS TUS_ISIK	DW_SKAIS2. LOG_PE_SOODUST US_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_STAAZ	DW_SKAIS2. LOG_PE_STAAZ	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_STAAZ_A RVUTATUD	DW_SKAIS2. LOG_PE_STAAZ_A RVUTATUD	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_STAAZ_D OKUMENT	DW_SKAIS2. LOG_PE_STAAZ_D OKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_STAAZ_IS IK	DW_SKAIS2. LOG_PE_STAAZ_IS IK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_PE_STAAZ_RI DA	DW_SKAIS2. LOG_PE_STAAZ_RI DA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_EMATA_S OTSMAS	DW_SKAIS2. LOG_SM_EMATA_S OTSMAS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_SOTSMA KS_AASTA	DW_SKAIS2. LOG_SM_SOTSMA KS_AASTA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_SOTSMA KS_AASTA_RIDA	DW_SKAIS2. LOG_SM_SOTSMA KS_AASTA_RIDA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_SOTSMA KS_ISIKUSTATUD	DW_SKAIS2. LOG_SM_SOTSMA KS_ISIKUSTATUD	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_SOTSMA KS_ISIKUSTATUD_ EM	DW_SKAIS2. LOG_SM_SOTSMA KS_ISIKUSTATUD_ EM	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_SOTSMA KS_LIIK_REEGEL	DW_SKAIS2. LOG_SM_SOTSMA KS_LIIK_REEGEL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SM_SOTSMA KS_SKEEM	DW_SKAIS2. LOG_SM_SOTSMA KS_SKEEM	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SY_SEDEL	DW_SKAIS2. LOG_SY_SEDEL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SY_SEDEL_T EKST	DW_SKAIS2. LOG_SY_SEDEL_T EKST	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_SY_TEADE	DW_SKAIS2. LOG_SY_TEADE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_TE_TEENUS	DW_SKAIS2. LOG_TE_TEENUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_TE_TEENUS_ OSUTAMINE	DW_SKAIS2. LOG_TE_TEENUS_ OSUTAMINE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_TM_MAATRIKS	DW_SKAIS2. LOG_TM_MAATRIKS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')
SKAIS2. LOG_TM_MAATRIK S_TINGIMUS	DW_SKAIS2. LOG_TM_MAATRIK S_TINGIMUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('([LAST_LOAD_DT]','yyyy-mm-dd hh24:mi:ss')

SKAIS2. LOG_TM_TINGIMU S_ALUS	DW_SKAIS2. LOG_TM_TINGIMUS _ALUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TM_TINGIMU S_KONTEKST	DW_SKAIS2. LOG_TM_TINGIMUS _KONTEKST	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TM_TINGIMU S_RIDA	DW_SKAIS2. LOG_TM_TINGIMUS _RIDA	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TM_TINGIMU S_VARIANT	DW_SKAIS2. LOG_TM_TINGIMUS _VARIANT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TM_TINGIMU S_VEERG	DW_SKAIS2. LOG_TM_TINGIMUS _VEERG	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TRY_TRYKIS	DW_SKAIS2. LOG_TRY_TRYKIS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TRY_TRYKIS_ ELEMENT	DW_SKAIS2. LOG_TRY_TRYKIS_ ELEMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TRY_TRYKIS_ NOUE_REEGEL	DW_SKAIS2. LOG_TRY_TRYKIS_ NOUE_REEGEL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TRY_TRYKIS_ REEGEL	DW_SKAIS2. LOG_TRY_TRYKIS_ REEGEL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TR_PEATAMI NE	DW_SKAIS2. LOG_TR_PEATAMI NE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_TR_TOOTAMI NE	DW_SKAIS2. LOG_TR_TOOTAMI NE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANNE	DW_SKAIS2. LOG_YL_YLESANNE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANN E_DOKUMENT	DW_SKAIS2. LOG_YL_YLESANN E_DOKUMENT	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANN E_ISIK	DW_SKAIS2. LOG_YL_YLESANN E_ISIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANN E_LIIK	DW_SKAIS2. LOG_YL_YLESANN E_LIIK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANN E_LIIK_ROLL	DW_SKAIS2. LOG_YL_YLESANN E_LIIK_ROLL	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANN E_OLEK	DW_SKAIS2. LOG_YL_YLESANN E_OLEK	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LOG_YL_YLESANN E_SEOS	DW_SKAIS2. LOG_YL_YLESANN E_SEOS	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_LISAM_AEG >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. LX_LIIDES_VAART US_VASTAVUS	DW_SKAIS2. LX_LIIDES_VAART US_VASTAVUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2.MT_MALL	DW_SKAIS2. MT_MALL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. MT_MALL_TEKST	DW_SKAIS2. MT_MALL_TEKST	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PA_PARAMEETER	DW_SKAIS2. PA_PARAMEETER	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PA_VAARTUS	DW_SKAIS2. PA_VAARTUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PA_VAARTUS_M	DW_SKAIS2. PA_VAARTUS_M	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT'),'yyyy-mm-dd hh24:mi:ss')	

SKAIS2. PE_MUU_PENSION	DW_SKAIS2. PE_MUU_PENSION	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_MUU_PENSION _DOKUMENT	DW_SKAIS2. PE_MUU_PENSION _DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_PENSIONILISA	DW_SKAIS2. PE_PENSIONILISA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_PENSIONILISA_ DOKUMENT	DW_SKAIS2. PE_PENSIONILISA_ DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_PENSIONILISA_ ISIK	DW_SKAIS2. PE_PENSIONILISA_ ISIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_SOODUSTUS	DW_SKAIS2. PE_SOODUSTUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_SOODUSTUS_ DOKUMENT	DW_SKAIS2. PE_SOODUSTUS_ DOKUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_SOODUSTUS_ ISIK	DW_SKAIS2. PE_SOODUSTUS_ ISIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2.PE_STAAZ	DW_SKAIS2. PE_STAAZ	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_STAAZ_ARVUT ATUD	DW_SKAIS2. PE_STAAZ_ARVUT ATUD	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_STAAZ_DOKUM ENT	DW_SKAIS2. PE_STAAZ_DOKUM ENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_STAAZ_ISIK	DW_SKAIS2. PE_STAAZ_ISIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. PE_STAAZ_RIDA	DW_SKAIS2. PE_STAAZ_RIDA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_EMMA_SOTSM AKS	DW_SKAIS2. SM_EMMA_SOTSM AKS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_SOTSMMAKS_A ASTA	DW_SKAIS2. SM_SOTSMMAKS_AA STA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_SOTSMMAKS_A ASTA_RIDA	DW_SKAIS2. SM_SOTSMMAKS_AA STA_RIDA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_SOTSMMAKS_ISI KUSTATUD	DW_SKAIS2. SM_SOTSMMAKS_ISI KUSTATUD	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_SOTSMMAKS_ISI KUSTATUD_EMMA	DW_SKAIS2. SM_SOTSMMAKS_ISI KUSTATUD_EMMA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_SOTSMMAKS_LII K_REEGEL	DW_SKAIS2. SM_SOTSMMAKS_LII K_REEGEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SM_SOTSMMAKS_S KEEM	DW_SKAIS2. SM_SOTSMMAKS_SK EEM	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SQLN_EXPLAIN_PL AN	DW_SKAIS2. SQLN_EXPLAIN_PL AN	REWRITE		
SKAIS2.SY_SEDEL	DW_SKAIS2. SY_SEDEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. SY_SEDEL_TEKST	DW_SKAIS2. SY_SEDEL_TEKST	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2.SY_TEADE	DW_SKAIS2. SY_TEADE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TE_TEENUS	DW_SKAIS2. TE_TEENUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	

SKAIS2. TE_TEENUS_OSUT AMINE	DW_SKAIS2. TE_TEENUS_OSUT AMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_MAATRIKS	DW_SKAIS2. TM_MAATRIKS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_MAATRIKS_TIN GIMUS	DW_SKAIS2. TM_MAATRIKS_TIN GIMUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_TINGIMUS_ALUS	DW_SKAIS2. TM_TINGIMUS_ALUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_TINGIMUS_KO NTEKST	DW_SKAIS2. TM_TINGIMUS_KO NTEKST	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_TINGIMUS_RIDA	DW_SKAIS2. TM_TINGIMUS_RIDA	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_TINGIMUS_VA RIANT	DW_SKAIS2. TM_TINGIMUS_VA RIANT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TM_TINGIMUS_VEE RG	DW_SKAIS2. TM_TINGIMUS_VEE RG	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TRY_TRYKIS	DW_SKAIS2. TRY_TRYKIS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TRY_TRYKIS_ELE MENT	DW_SKAIS2. TRY_TRYKIS_ELE MENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TRY_TRYKIS_NOU E_REEGEL	DW_SKAIS2. TRY_TRYKIS_NOU E_REEGEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TRY_TRYKIS_REE GEL	DW_SKAIS2. TRY_TRYKIS_REE GEL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TR_PEATAMINE	DW_SKAIS2. TR_PEATAMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. TR_TOOTAMINE	DW_SKAIS2. TR_TOOTAMINE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE	DW_SKAIS2. YL_YLESANNE	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE_DO KUMENT	DW_SKAIS2. YL_YLESANNE_DO KUMENT	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE_ISIK	DW_SKAIS2. YL_YLESANNE_ISIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE_LIIK	DW_SKAIS2. YL_YLESANNE_LIIK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE_LII K_ROLL	DW_SKAIS2. YL_YLESANNE_LIIK _ROLL	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE_OL EK	DW_SKAIS2. YL_YLESANNE_OL EK	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS2. YL_YLESANNE_SE OS	DW_SKAIS2. YL_YLESANNE_SE OS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_ISIK. INCOME_TAX_EXE MPTION	DW_SKAIS_ISIK. INCOME_TAX_EXE MPTION	LOAD_MODIFIED	(case when coalesce(LAST_MODIFIED_DATE, CREATED_DATE) < DATE '1900-01-01' then DATE '1900-01-01' else coalesce(LAST_MODIFIED_DATE, CREATED_DATE) end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_ISIK. LOG_INCOME_TAX _EXEMPTION	DW_SKAIS_ISIK. LOG_INCOME_TAX _EXEMPTION	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_ISIK. LOG_REQUISITE	DW_SKAIS_ISIK. LOG_REQUISITE	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_ISIK. REQUISITE	DW_SKAIS_ISIK. REQUISITE	REWRITE		

SKAIS_NOTIFICATION.LOG_MESSAGE	DW_SKAIS_NOTIFICATION.LOG_MESSAGE	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.LOG_MESSAGE_ATTACHMENT	DW_SKAIS_NOTIFICATION.LOG_MESSAGE_ATTACHMENT	LOAD_MODIFIED_WOID	LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.LOG_MESSAGE_I18N	DW_SKAIS_NOTIFICATION.LOG_MESSAGE_I18N	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.LOG_MESSAGE_SUBJECT	DW_SKAIS_NOTIFICATION.LOG_MESSAGE_SUBJECT	LOAD_MODIFIED_WOID	LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.LOG_NOTIFICATION	DW_SKAIS_NOTIFICATION.LOG_NOTIFICATION	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.MESSAGE	DW_SKAIS_NOTIFICATION.MESSAGE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.MESSAGE_ATTACHMENT	DW_SKAIS_NOTIFICATION.MESSAGE_ATTACHMENT	LOAD_MODIFIED	(case when coalesce(LAST_MODIFIED_DATE, CREATED_DATE) < DATE '1900-01-01' then DATE '1900-01-01' else coalesce(LAST_MODIFIED_DATE, CREATED_DATE) end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.MESSAGE_I18N	DW_SKAIS_NOTIFICATION.MESSAGE_I18N	LOAD_MODIFIED	(case when coalesce(LAST_MODIFIED_DATE, CREATED_DATE) < DATE '1900-01-01' then DATE '1900-01-01' else coalesce(LAST_MODIFIED_DATE, CREATED_DATE) end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_NOTIFICATION.MESSAGE_SUBJECT	DW_SKAIS_NOTIFICATION.MESSAGE_SUBJECT	REWRITE		
SKAIS_NOTIFICATION.NOTIFICATION	DW_SKAIS_NOTIFICATION.NOTIFICATION	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.COMBINED_OFFERING	DW_SKAIS_PAKKUMUS.COMBINED_OFFERING	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.COMBINED_OFFERING_ATTACHMENT	DW_SKAIS_PAKKUMUS.COMBINED_OFFERING_ATTACHMENT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_COMBINED_OFFERING	DW_SKAIS_PAKKUMUS.LOG_COMBINED_OFFERING	LOAD_MODIFIED_WOID	LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_COMBINED_OFFERING_ATTACHMENT	DW_SKAIS_PAKKUMUS.LOG_COMBINED_OFFERING_ATTACHMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_OFFERING	DW_SKAIS_PAKKUMUS.LOG_OFFERING	LOAD_MODIFIED_WOID	LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_OFFERING_BENEFIT	DW_SKAIS_PAKKUMUS.LOG_OFFERING_BENEFIT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_OFFERING_GROUP	DW_SKAIS_PAKKUMUS.LOG_OFFERING_GROUP	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_OFFERING_SUBJECT	DW_SKAIS_PAKKUMUS.LOG_OFFERING_SUBJECT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	
SKAIS_PAKKUMUS.LOG_OFFERING_SUBJECT_DETAIL	DW_SKAIS_PAKKUMUS.LOG_OFFERING_SUBJECT_DETAIL	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')	

SKAIS_PAKKUMUS.OFFERING	DW_SKAIS_PAKKUMUS.OFFERING	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PAKKUMUS.OFFERING_ADDITIONAL_INFO	DW_SKAIS_PAKKUMUS.OFFERING_ADDITIONAL_INFO	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PAKKUMUS.OFFERING_BENEFIT	DW_SKAIS_PAKKUMUS.OFFERING_BENEFIT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PAKKUMUS.OFFERING_CALCULATION_INFO	DW_SKAIS_PAKKUMUS.OFFERING_CALCULATION_INFO	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PAKKUMUS.OFFERING_GROUP	DW_SKAIS_PAKKUMUS.OFFERING_GROUP	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PAKKUMUS.OFFERING_SUBJECT	DW_SKAIS_PAKKUMUS.OFFERING_SUBJECT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PAKKUMUS.OFFERING_SUBJECT_DETAIL	DW_SKAIS_PAKKUMUS.OFFERING_SUBJECT_DETAIL	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.ALLOWANCE	DW_SKAIS_PEREHYVITIS.ALLOWANCE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.ALLOWANCE_SUBJECT	DW_SKAIS_PEREHYVITIS.ALLOWANCE_SUBJECT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.CALCULATED_ALLOWANCE	DW_SKAIS_PEREHYVITIS.CALCULATED_ALLOWANCE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.CALCULATED_ALLOWANCE_PAYMENT	DW_SKAIS_PEREHYVITIS.CALCULATED_ALLOWANCE_PAYMENT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.CALCULATED_ALLOWANCE_SUBTYPE	DW_SKAIS_PEREHYVITIS.CALCULATED_ALLOWANCE_SUBTYPE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.DECISION	DW_SKAIS_PEREHYVITIS.DECISION	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.DECISION_BASIS	DW_SKAIS_PEREHYVITIS.DECISION_BASIS	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.DECISION_BASIS_SETTING	DW_SKAIS_PEREHYVITIS.DECISION_BASIS_SETTING	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.DECISION_EVALUATION_RESULT	DW_SKAIS_PEREHYVITIS.DECISION_EVALUATION_RESULT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.EVALUATION_RESULT	DW_SKAIS_PEREHYVITIS.EVALUATION_RESULT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.LOG_ALLOWANCE	DW_SKAIS_PEREHYVITIS.LOG_ALLOWANCE	LOAD_MODIFIED_WOIID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.LOG_ALLOWANCE_SUBJECT	DW_SKAIS_PEREHYVITIS.LOG_ALLOWANCE_SUBJECT	LOAD_MODIFIED_WOIID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.LOG_CALCULATED_ALLOWANCE	DW_SKAIS_PEREHYVITIS.LOG_CALCULATED_ALLOWANCE	LOAD_MODIFIED_WOIID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')

SKAIS_PEREHYVITIS. LOG_CALCULATED_ALLOWANCE_PAY ME	DW_SKAIS_PEREHYVITIS. LOG_CALCULATED_ALLOWANCE_PAY ME	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS. LOG_CALCULATED_ALLOWANCE_SU BTY	DW_SKAIS_PEREHYVITIS. LOG_CALCULATED_ALLOWANCE_SU BTY	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.LOG_DECISION	DW_SKAIS_PEREHYVITIS. LOG_DECISION	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS. LOG_DECISION_BA SIS	DW_SKAIS_PEREHYVITIS. LOG_DECISION_BA SIS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS. LOG_DECISION_BA SIS_SETTING	DW_SKAIS_PEREHYVITIS. LOG_DECISION_BA SIS_SETTING	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS. LOG_DECISION_EV ALUATION_RESULT	DW_SKAIS_PEREHYVITIS. LOG_DECISION_EV ALUATION_RESULT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS. LOG_EVALUATION _RESULT	DW_SKAIS_PEREHYVITIS. LOG_EVALUATION _RESULT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.LOG_OFFERING	DW_SKAIS_PEREHYVITIS. LOG_OFFERING	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS. LOG_PARAPHRASE	DW_SKAIS_PEREHYVITIS. LOG_PARAPHRASE	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.OFFERING	DW_SKAIS_PEREHYVITIS.OFFERING	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PEREHYVITIS.PARAPHRASE	DW_SKAIS_PEREHYVITIS. PARAPHRASE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING.ALLOWANCE	DW_SKAIS_PROCEEDING. ALLOWANCE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING. ALLOWANCE_STAT US	DW_SKAIS_PROCEEDING. ALLOWANCE_STAT US	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING. ALLOWANCE_TRA NSITION_EXCLUSI ON	DW_SKAIS_PROCEEDING. ALLOWANCE_TRAN SITION_EXCLUSION	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING. ALLOWANCE_TRA NSITION_STATE	DW_SKAIS_PROCEEDING. ALLOWANCE_TRAN SITION_STATE	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING.DECISION	DW_SKAIS_PROCEEDING.DECISION	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING. LOG_ALLOWANCE	DW_SKAIS_PROCEEDING. LOG_ALLOWANCE	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING. LOG_ALLOWANCE _STATUS	DW_SKAIS_PROCEEDING. LOG_ALLOWANCE_ STATUS	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')
SKAIS_PROCEEDING. LOG_ALLOWANCE _TRANSITION_EXC LU	DW_SKAIS_PROCEEDING. LOG_ALLOWANCE_ TRANSITION_EXCLU	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss')

SKAIS_PROCEEDING.LOG_ALLOWANCE_TRANSITION_STATE	DW_SKAIS_PROCEDING.LOG_ALLOWANCE_TRANSITION_STATE	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.LOG_DECISION	DW_SKAIS_PROCEDING.LOG_DECISION	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.LOG_PAYMENT	DW_SKAIS_PROCEDING.LOG_PAYMENT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.LOG_PAYMENT_CHANNEL	DW_SKAIS_PROCEDING.LOG_PAYMENT_CHANNEL	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.LOG_SUBJECT	DW_SKAIS_PROCEDING.LOG_SUBJECT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.PAYMENT	DW_SKAIS_PROCEDING.PAYMENT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.PAYMENT_CHANNEL	DW_SKAIS_PROCEDING.PAYMENT_CHANNEL	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_PROCEEDING.SUBJECT	DW_SKAIS_PROCEDING.SUBJECT	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.ISO	DW_SKAIS_TECHNICAL_AID.ISO	REWRITE		
SKAIS_TECHNICAL_AID.ISO_CONDITION	DW_SKAIS_TECHNICAL_AID.ISO_CONDITION	REWRITE		
SKAIS_TECHNICAL_AID.ISO_CONDITION_CODE	DW_SKAIS_TECHNICAL_AID.ISO_CONDITION_CODE	REWRITE		
SKAIS_TECHNICAL_AID.ISO_JUSTIFICATION_LIMIT	DW_SKAIS_TECHNICAL_AID.ISO_JUSTIFICATION_LIMIT	REWRITE		
SKAIS_TECHNICAL_AID.JUSTIFICATION	DW_SKAIS_TECHNICAL_AID.JUSTIFICATION	REWRITE		
SKAIS_TECHNICAL_AID.LOG_ISO	DW_SKAIS_TECHNICAL_AID.LOG_ISO	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_ISO_CONDITION	DW_SKAIS_TECHNICAL_AID.LOG_ISO_CONDITION	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_ISO_CONDITION_CODE	DW_SKAIS_TECHNICAL_AID.LOG_ISO_CONDITION_CODE	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_ISO_JUSTIFICATION_LIMIT	DW_SKAIS_TECHNICAL_AID.LOG_ISO_JUSTIFICATION_LIMIT	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_JUSTIFICATION	DW_SKAIS_TECHNICAL_AID.LOG_JUSTIFICATION	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_PROVIDER	DW_SKAIS_TECHNICAL_AID.LOG_PROVIDER	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_RENT_PERIOD	DW_SKAIS_TECHNICAL_AID.LOG_RENT_PERIOD	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	
SKAIS_TECHNICAL_AID.LOG_SPECIAL_REQUEST	DW_SKAIS_TECHNICAL_AID.LOG_SPECIAL_REQUEST	LOAD_MODIFIED_WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT','YYYY-MM-DD HH24:MI:SS')	

SKAIS_TECHNICAL AID. LOG_TRANSACTION	DW_SKAIS_TECHNI CALAID. LOG_TRANSACTION	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AID. LOG_TRANSACTION N_ROW	DW_SKAIS_TECHNI CALAID. LOG_TRANSACTION N_ROW	LOAD_MODIFIED_ WOID	LOG_ACTION in ('D','I') and LOG_CREATE_TIME >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AID.PROVIDER	DW_SKAIS_TECHNI CALAID.PROVIDER	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AID.RENT_PERIOD	DW_SKAIS_TECHNI CALAID. RENT_PERIOD	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AID. SPECIAL_REQUEST	DW_SKAIS_TECHNI CALAID. SPECIAL_REQUEST	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AID.TRANSACTION	DW_SKAIS_TECHNI CALAID. TRANSACTION	LOAD_MODIFIED	(case when LAST_MODIFIED_DATE < DATE '1900-01-01' then DATE '1900-01-01' else LAST_MODIFIED_DATE end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AID. TRANSACTION_ROW	DW_SKAIS_TECHNI CALAID. TRANSACTION_ROW	LOAD_MODIFIED	(case when coalesce(LAST_MODIFIED_DATE, CREATED_DATE) < DATE '1900-01-01' then DATE '1900-01-01' else coalesce(LAST_MODIFIED_DATE, CREATED_DATE) end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_TECHNICAL AIDXROAD. XROAD_INBOUND_ REQUEST	DW_SKAIS_TECHNI CALAIDXROAD. XROAD_INBOUND_ REQUEST	LOAD_MODIFIED	(case when REQUEST_RECEIVED < DATE '1900-01-01' then DATE '1900-01-01' else REQUEST_RECEIVED end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS2VALINE. SKAIS1_PRT_DATA	DW_SKAIS2_VALIN E_ODS. SKAIS1_PRT_DATA	REWRITE		
SKAIS_RINAFACADE	CASE_INFO	LOAD_MODIFIED	(case when coalesce(LAST_MODIFIED_DATE, CREATED_DATE) < DATE '1900-01-01' then DATE '1900-01-01' else coalesce(LAST_MODIFIED_DATE, CREATED_DATE) end) >=TO_DATE('LAST_LOAD_DT'],'yyyy-mm-dd hh24:mi:ss')	
SKAIS_RINAFACADE	CASE_STATE	LOAD_MODIFIED	(case when CREATED_DATE < DATE "1900-01-01" then DATE "1900-01-01" else CREATED_DATE end) >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
SKAIS_RINAFACADE	LOG_CASE_INFO	LOAD_MODIFIED_ WOID	LOG_ACTION in ("D","I") and LOG_CREATE_TIME >=TO_DATE ("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
SKAIS_RINAFACADE	LOG_CASE_STATE	LOAD_MODIFIED_ WOID	LOG_ACTION in ("D","I") and LOG_CREATE_TIME >=TO_DATE ("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
SKAIS2.IS_KONTAK T_STAATUS	DW_SKAIS2. IS_KONTAKT_STAA TUS	LOAD_MODIFIED	(case when MUUTM_AEG < DATE "1900-01-01" then DATE "1900-01-01" else MUUTM_AEG end) >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
SKAIS2.LOG_IS_KO NTAKT_STAATUS	DW_SKAIS2. LOG_IS_KONTAKT_ STAATUS	LOAD_MODIFIED_ WOID	LOG_ACTION in ("D","I") and LOG_LISAM_AEG >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
skais_doctor_assess ment.activity_catego ry	DW_SKAIS_DOCTO R_ASSESSMENT_O DS.activity_category	LOAD_MODIFIED	(case when MUUTM_AEG < DATE "1900-01-01" then DATE "1900-01-01" else MUUTM_AEG end) >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
skais_doctor_assess ment. domain_category	DW_SKAIS_DOCTO R_ASSESSMENT_O DS.domain_category	LOAD_MODIFIED	(case when MUUTM_AEG < DATE "1900-01-01" then DATE "1900-01-01" else MUUTM_AEG end) >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
skais_doctor_assess ment.exp_deviation_ icf_category	DW_SKAIS_DOCTO R_ASSESSMENT_O DS. exp_deviation_icf_ca tegrity	LOAD_MODIFIED	(case when MUUTM_AEG < DATE "1900-01-01" then DATE "1900-01-01" else MUUTM_AEG end) >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
skais_doctor_assess ment. log_activity_category	DW_SKAIS_DOCTO R_ASSESSMENT_O DS. log_activity_category	LOAD_MODIFIED_ WOID	LOG_ACTION in ("D","I") and LOG_LISAM_AEG >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
skais_doctor_assess ment. log_domain_category	DW_SKAIS_DOCTO R_ASSESSMENT_O DS. log_domain_category	LOAD_MODIFIED_ WOID	LOG_ACTION in ("D","I") and LOG_LISAM_AEG >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	
skais_doctor_assess ment.log_exp_deviat ion_icf_category	DW_SKAIS_DOCTO R_ASSESSMENT_O DS. log_exp_deviation_ic f_category	LOAD_MODIFIED_ WOID	LOG_ACTION in ("D","I") and LOG_LISAM_AEG >=TO_DATE("LAST_LOAD_DT"],"yyyy-mm-dd hh24:mi:ss")	

SKA SKAIS2 DWH Pentaho Data-Intergration paigaldusjuhend

Paigaldus üldjuhul sama, nagu Skais1 puhul. Eristab ainult pentaho repositooriumi ehk laadimiskriptide ja ühendusfailide sisu.

Pentaho Data Integration - kettle

Pentaho Data Integration on ETL tarkvara – andmete eksport, töötlemine ja ümberlaadimine.

Paigaldatakse Pentaho Community Edition (CE), mis on vabavaraline tarkvara

Kasutaja

Pentaho java protsess käivitatakse pentaho kasutaja alt.

```
sudo useradd pentaho
```

```
sudo passwd pentaho
```

Java JDK

Pentaho jaoks paigaldatakse uusim JDK8 versioon

```
RPM installi puhul paigaldatakse JDK asukohta /usr/java/default
```

```
cd /tmp/java
```

```
sudo rpm -Uvh jdk-8u151-linux-x64.rpm
```

Lisada JAVA_HOME keskkonnamuutuja pentaho kasutajale

```
vi /home/pentaho/.bash_profile
```

```
export JAVA_HOME=/usr/java/default/jre
```

Install

PDI ei vaja eraldi paigaldust. Arhiiv lahti pakkida ja õigused anda.

Pentaho Data Integration CE versiooni saab alla laadida sourceforge-st:

<https://sourceforge.net/projects/pentaho/files/Data%20Integration/>

Pentaho alla laadida, lahti pakkida ja õigused sättida.

```
cd /tmp/pentaho
```

```
unzip pdi-ce-8.0.zip
```

```
sudo mkdir /opt/data-integration
```

```
sudo mv * /opt/data-integration
```

```
cd /opt/data-integration
```

```
sudo chmod -R u+x *.sh
```

```
sudo chown -R pentaho *
```

JDBC andmeühendused

Pentaho on java baseeruv ja seega kasutab andmebaasi ühendusteks JDBC ühendusi.

Pentaho ühendub oracle ja vertica andmebaasiga.

Andmebaasi draiverid

Paigaldada vertica ja oracle sobilikud andmebaasi draiverid.

Pentaho nimekiri draiveritest:

<https://help.pentaho.com/Documentation/5.3/OD0/160/010>

Vertica 9.0.0 draiveri leiab:

<https://my.vertica.com/download/vertica/client-drivers/>

Oracle draiveri versioon valida vastavalt andmebaasimootori versioonile

<http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html>

Draiverite jar failid liigutada pentaho lib kausta

`/opt/pentaho/data-integration/lib`

Pentaho repositoorium

Kirjeldada pentaho repositoorium kus hakkavad laadimisskriptid asuma

`vi /home/pentaho/.kettle/repositories.xml`

Jälgida asukohta – asukoht on osades skriptides sisse kirjutatud

```
<?xml version="1.0" encoding="UTF-8"?>
<repositories>
  <repository>  <id>KettleFileRepository</id>
    <name>ska</name>
    <description>SKAIS2 DWH REPO</description>
    <is_default>false</is_default>
    <base_directory>/home/pentaho/repo_skais2</base_directory>
    <read_only>N</read_only>
    <hides_hidden_files>N</hides_hidden_files>
  </repository> </repositories>
```

Luu repositooriumi kaust

`sudo mkdir /home/pentaho/repo_skais2`

`sudo chown pentaho:pentaho /home/pentaho/repo_skais2`

Repo olemasolu saab kontrollida:

`cd /srv/pentaho`

`./kitchen.sh -listrep`

Laadimisskriptid

Kopeerida pentaho failid (laadimisskriptid) pentaho repositooriumi kausta.

```
cp -R /tmp/laadimisskriptid/* /home/pentaho/repo
```

```
sudo chown -R pentaho:pentaho /home/pentaho/repo
```

Andmeühenduste failid

Pentaho repositooriumis asuvad andmeühendusi defineerivad kdb failid. Ühenduse failides tuleb muuta parameetrid live Vertica-le vastavaks

Krüpteerida vertica kasutaja „pentaho“ parool ja SKAIS andmebaasi ühenduse kasutaja. Salvesta väljund, mis on kujul „Encrypted 123456...“

```
sudo su - pentaho
```

```
/opt/data-integration/encr.sh -kettle vertica_voi_oracle_pentaho_parool
```

Conn_dwh on vertica ühendust kirjeldav fail

```
vi /home/pentaho/repo/conn_dwh.kdb
```

Muuta vertica ühendusel serverinimi, baasinimi, port, parool(pentaho Vertica kasutaja). Port on kirjeldatud kahes kohas. Nimi conn_ska_wrh peab samaks jääma

```
<connection>
<name>conn_ska_wrh</name>
<server>ska-wrh-1a.prd.tehik.ee</server>
<type>VERTICA5</type>
<access>Native</access>
<database>ska_wrh_db</database>
<port>5433</port>
<username>pentaho</username>
<password>Encrypted *****</password>
<servername/>
<data_tablespace/>
<index_tablespace/>
<attributes>
.....
<attribute><code>PORT_NUMBER</code><attribute>5433</attribute></attribute>
```

Conn_oracle on Oracle Skais2 baasi ühendust kirjeldav fail

```
vi /home/pentaho/repo/conn_oracle.kdb
```

```
<connection>
<name>conn_oracle</name>
<server/>
<type>GENERIC</type>
<access>Native</access>
<database/>
<port>1521</port>
<username>SKA_ANDMEAIT</username>
<password>*****</password>
<servername/>
<data_tablespace/>
<index_tablespace/>
<attributes>
<attribute><code>CUSTOM_DRIVER_CLASS</code><attribute>oracle.jdbc.driver.OracleDriver</attribute></attribute>
<attribute><code>CUSTOM_URL</code><attribute>jdbc:oracle:thin:@//84.50.231.112:1521/PSKAIS2APP</attribute></attribute>
<attribute><code>FORCE_IDENTIFIERS_TO_LOWERCASE</code><attribute>N</attribute></attribute>
<attribute><code>FORCE_IDENTIFIERS_TO_UPPERCASE</code><attribute>N</attribute></attribute>
<attribute><code>IS_CLUSTERED</code><attribute>N</attribute></attribute>
<attribute><code>PORT_NUMBER</code><attribute>1521</attribute></attribute>
<attribute><code>PRESERVE_RESERVED_WORD_CASE</code><attribute>Y</attribute></attribute>
<attribute><code>QUOTE_ALL_FIELDS</code><attribute>N</attribute></attribute>
<attribute><code>SUPPORTS_BOOLEAN_DATA_TYPE</code><attribute>Y</attribute></attribute>
<attribute><code>SUPPORTS_TIMESTAMP_DATA_TYPE</code><attribute>Y</attribute></attribute>
<attribute><code>USE_POOLING</code><attribute>N</attribute></attribute>
</attributes>
</connection>
```

SKA SKAIS2 DWH Vertica ühendumise juhend

Keskkondade parameetrid

Vaata parameetreid lehelt [Serverid ja ühendused](#)

Kasutaja

Skais vertica kasutajaid haldab Tehik (itabi)

JDBC

Java põhiste klientide kaudu ühendumine Dbeaver näitel

Driver

Laadi alla Vertica JDBC uusima versiooni draiveri fail ja paiguta oma arvutis kindlasse kohta.

<https://www.vertica.com/download/vertica/client-drivers/>

https://www.vertica.com/client_drivers/24.2.x/24.2.0-0/vertica-jdbc-24.2.0-0.jar

Dbeaver

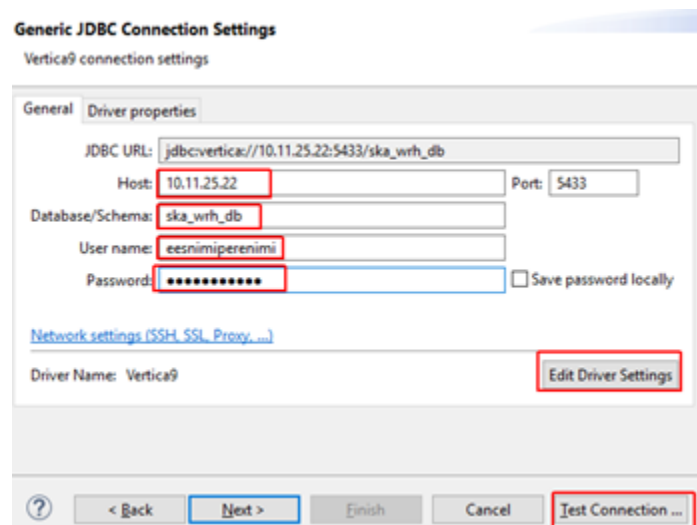
Laadi alla ja paigalda Dbeaver Community edition (koos JRE-ga)

<https://dbeaver.io/download/>

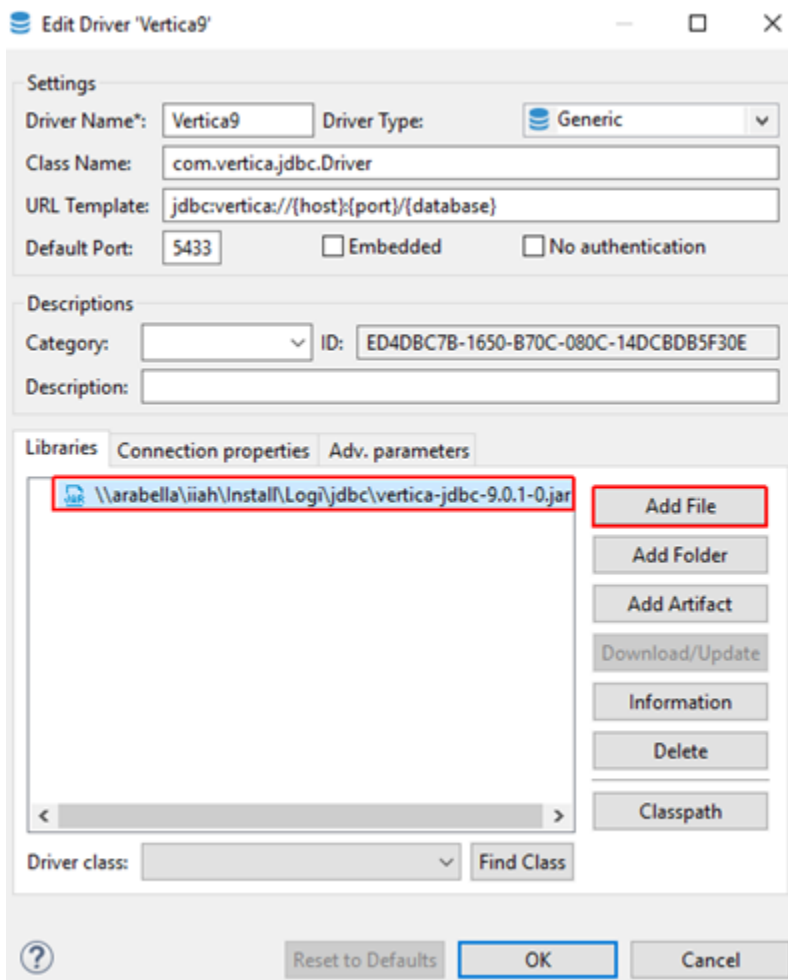
Dbeaver ühendus

Kui dbeaver on käivitatud, siis tuleb luua uus ühendus, määrata draiver ja sättida parameetrid.

New Connection > Connection Type Vertica > Määrata ühenduse parameetrid



Edit Driver Settings alt lisada eelnevalt allalaetud vertica-jdbc-***.jar fail



Test connection

Ühendu lisatud ühenduse andmebaasiga

ODBC

ODBC põhiste klientide kaudu ühendamine

Client

Tõmba alla ja paigalda Vertica uusim Windows Client

https://www.vertica.com/client_drivers/24.2.x/24.2.0-0/VerticaSetup-24.2.0-0.exe

ODBC

Lisada ODBC ühendus vastava baasi parameetritega

Start > ODBC Data Sources (64-bit) > Add > Vertica

Määra Database, Server, Port ja kasutaja Basic Settings alt.

HPE Vertica ODBC DSN Configuration

Basic Settings Client Settings Server Settings About

Connection info

DSN name *:

Description:

Database *:

Server *:

Backup servers:

Port:

Use connection load balancing:

Authentication

User name *:

Password:

Prompt for missing password:

Use Windows authentication:

SKA SKAIS2 DW süsteemsed tabelid

COLUMNS SKAIS2

Väljade/veergude andmed

Välja nimi	Välja tüüp	Selgitus	Andmete näited
COLUMN_VERSION_ID	int	Surrogaatvõti	6750
DATA_SOURCE_CODE	varchar (20)	Andmeallika/lähtesüsteemi kood	'SKAIS2'
SOURCE_SCHEMA_NAME	varchar (128)	Skeemi nimi lähtesüsteemis	'SKAIS2'
SOURCE_TABLE_NAME	varchar (128)	Tabeli nimi lähtesüsteemis	'AR_PROTSESS_VEATEADE'
SOURCE_COLUMN_NAME	varchar (128)	Välja/veeru nimi lähtesüsteemis	'VEATEADE_SYS'
DWH_COLUMN_NAME	varchar (128)	Välja/veeru nimi andmelaos	'VEATEADE_SYS'
SOURCE_COLUMN_COMMENT	varchar (4000)	Välja/veeru kirjeldus lähtesüsteemis	'Kommentaar'
DWH_COLUMN_COMMENT	varchar (4000)	Välja/veeru kirjeldus andmelaos	'Kommentaar'
DATA_TYPE	varchar (128)	Andmetüüp lähtesüsteemis	'VARCHAR2', 'INTEGER', 'DECIMAL', 'CLOB'
DATA_LENGTH	int	Andmevälja pikkus lähtesüsteemis	4000
DATA_PRECISION	int	Andmevälja täpsus lähtesüsteemis	18
DATA_SCALE	int	Andmevälja täpsus peale koma lähtesüsteemis	0
DWH_DATA_TYPE	varchar (255)	Andmetüüp andmelaos	'VARCHAR(20000)'
NULLABLE	char(1)	Kas veerg võib sisaldada puuduvaid (NULL) väärtusi (0/1)	1
IS_KEY	int	Kas veerg on primaarvõti (0/1)	0
LOAD_STATUS	varchar (20)	Välja/veeru laadimise staatus andmelaos	'LOAD' - Välja laadimine on sisse lülitatud 'NEW_LOAD' - Tegemist on uue väljaga (st haldur pole välja laadimise staatust üle vaadanud /kinnitanud), väli on laadimisse sisse lülitatud 'NEW_NOT_LOAD' - Tegemist on uue väljaga, väli ei ole laadimisse sisse lülitatud 'NOT_LOAD' - välja laadimine ei ole sisse lülitatud 'NO_OBJECT' - välja ei ole enam lähtesüsteemis (varem oli)
DWH_COLUMN_STATUS	varchar (30)	Välja staatus andmelaos:	'NEEDS_CREATION' - väli vajab loomist 'NEEDS_MODIFICATION' - väli vajab muutmist (näiteks lähteallikas on muutunud andmetüüp) 'READY' - väli on loodud/muudetud
SOURCE_COLUMN_ID	int	Välja/veeru ID lähtesüsteemis	7
IS_VALID	int	Kas tegemist on kehtiva kirje /versiooniga (0/1)	1
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2019-06-16 20:36:18'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2019-07-02 11:11:26'
MODIFIED_BY	varchar (32)	Kirje looja/muutja	'pentaho'

DATA_SOURCE_PARAMS SKAIS2

Lähtesüsteemi andmebaasi (tehnilised) parameetrid.

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(20)	Andmeallika/lähtesüsteemi kood	'SKAIS2'
DB_SERVER_NAME	varchar(32)	Serveri nimi või IP aadress	'test-test.ml.ee '
DB_PORT_NR	int	Pordi number	1234
DB_NAME	varchar(32)	Andmebaasi nimi	'SKAIS2'
DB_USERNAME	varchar(32)	Kasutajanimi	'SKAIS2_USER_NAME'
DB_PASSWORD	varchar(128)	Parooli räsi (luuakse pentaho encr.sh abil). Vt täpsemalt paigaldusjuhendist.	'Encrypted 0123456789ABCDEF'

DATA_SOURCES SKAIS2

Andmeallikad (lähte-andmebaasid)

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_VERSION_ID	int	Andmeallika versiooni ID. Surrogaatvõti.	1
DATA_SOURCE_CODE	varchar (32)	Andmeallika/lähtesüsteemi kood.	'SKAIS2'
DATA_SOURCE_NAME	varchar (128)	Lähtesüsteemi nimetus	'SKAIS2'
DATA_SOURCE_TYPE	varchar (20)	Lähtesüsteemi tüüp	'SOURCE_DB' - lähtesüsteemiks on andmebaas 'TARGET' - andmelao andmebaas
DBMS_TYPE	varchar (20)	Andmebaasi tüüp	'ORACLE'
IS_ACTIVE	int	Kas lähtesüsteemist andmete laadimine on aktiivne (võimaldab terve lähtesüsteemi laadimise korraga välja lülitada)	1
IS_LOAD_TABLES	int	Kas lähtesüsteemist laaditakse andmetabeleid (table)	0
IS_LOAD_VIEWS	int	Kas lähtesüsteemist laaditakse vaateid (view)	0
IS_LOAD_MVIEWS	int	Kas lähtesüsteemist laaditakse materialiseeritud vaateis (materialized view)	1
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	1
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2019-06-14 17:08:06'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2019-10-14 17:08:06', NULL
MODIFIED_BY	varchar (32)	Versiooni lisaja/muutja	'SYSTEM'
HAS_DEL_REC	int	Kas selles andmeallika korral kasutatakse kustutatud kirjete tühistamist.	0

DATA_TYPE_CONVERSIONS SKAIS2

Välja tüüpide teisendustabel

Välja nimi	Välja tüüp	Selgitus	Andmete näited
ID	int	Surrogaatvõti	2
DBMS_TYPE	varchar(20)	Lähtesüsteemi andmebaasi tüüp	'ORACLE'
DATA_TYPE	varchar(128)	Andmetüüp lähtesüsteemis	'DATE'
DATA_LENGTH	int	Andmevälja pikkus lähtesüsteemis	10
DATA_PRECISION	int	Andmevälja täpsus lähtesüsteemis	1
DATA_SCALE	int	Andmevälja täpsus peale koma lähtesüsteemis	1
TARGET_DATA_TYPE	varchar(128)	Andmetüüp (sh pikkus ja täpsus) andmelaos	'DATETIME'

DMARTS SKAIS2

Andmelehid. Kasutatakse andmelettide laadimise protsessis (hetkel seda SKAIS2 juures kasutusel ei ole).

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DMART_TABLE_ID	int	Surrogaatvõti	6750
DWH_SCHEMA_NAME	varchar(20)	Skeemi nimi, kus asub andmelett	'SKAIS2'
DMART_TABLE_NAME	varchar(255)	Andmeleti nimi	'PUUDED'
LOAD_ORDER_NR	int	Andmeleti laadimise järjekorranumber	1
IS_ACTIVE	int	Kas andmeleti laadimine on aktiivne (0/1)	1
DATA_SOURCE_CODE	varchar(30)	Andmeallika nimi	'SKAIS2'

DWH_SCHEMAS SKAIS2

Andmelao skeemide andmed.

Iga siin kirjeldatus skeemi kohta luuakse andmelaos vähemalt 2 skeemi (<skeemi nimi>_STG - vastava loogilise andmeallika *staging* tabelite skeem, <skeemi nimi>_ODS - vastava loogilise andmeallika ODS tabelite skeem). Lisaks luuakse vajaduse korral (kui HAS_DMARTS = 1) skeem <skeemi nimi>_DMART, mis sisaldab vastava andmeallika dimensiooni- ja faktitabeleid ja andmelette.

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DWH_SCHEMA_VERSION_ID	int	Skeemi versiooni ID	1
DWH_SCHEMA_NAME	varchar(32)	Andmelao skeemi/loogilise andmeallika nimi	'ORACLE'
DWH_SCHEMA_COMMENT	varchar(255)	Andmelao skeemi kirjeldus	'SKAIS2 põhiandmed'
DWH_SCHEMA_STATUS	varchar(30)	Andmelao skeemi staatus	'NEEDS_CREATION' - skeem on loomata 'NEEDS_MODIFICATION' - skeemis on objekte (tabeleid), mida on vaja muuta 'READY' - skeem ja kõik seal olevad objektid on loodud/muudetud
HAS_DMARTS	int	Kas skeem sisaldab andmelette	1
IS_ACTIVE	int	Kas skeemi/loogilise andmeallika andmete laadimine on aktiivne.	1
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	0
VALID_FROM	Timestamp	Kirje/versiooni kehtivuse algusaeg	'2019-06-14 17:08:06'
VALID_TO	Timestamp	Kirje/versiooni kehtivuse lõppaeg	'2019-06-16 20:50:15'
MODIFIED_BY	varchar(32)	Versiooni lisaja/muutja	'pentaho'

EVENT_LOG SKAIS2

Sündmuste logi

Välja nimi	Välja tüüp	Selgitus	Andmete näited
EVENT_ID	int	Sündmuse ID. Primaarvõti.	1
EVENT_DT	Timestamp	Sündmuse toimumise/registreerimise kuupäev ja kellaaeg	'2019-06-14 17:08:04'
EVENT_CLASS	varchar (20)	Sündmuse grupi kood. Võimalikud väärtused: INSTALLATION - installimine LOADING - laadimine	'INSTALLATION', 'LOADING'
EVENT_CODE	varchar (20)	Sündmuse tüüp. Võimalikud variandid: ERROR - viga, WARNING - hoiatus, SUCCESS - edukas tegevus, INFO - teade, START - installimise alustamine, END_COMPARE_META - metadata võrdlemise lõpp, INSERT DATA- andmete sisestamine, CREATE_TABLE - tabel loodud, SCHEMA CREATED - skeem loodud, COLUMN ADDED - veerg lisatud, TABLE CREATED - tabel loodud, END - installimise lõpp	'ERROR'
EVENT_MESSAGE	varchar (255)	Sündmuse kirjeldus	'ERROR in loading STG table'
RELATED_OBJECT	varchar (50)	Sündmusega seotud objekt (nt tabeli nimi, lähteallika nimi vms). Konkreetne objekt sõltub sündmuse koodist.	'ADM_SKAIS2.TABLES'
PROCESS_NAME	varchar (50)	Pentaho protsessi nimi, mis sündmuse tekitas	'run_upgrade'
STEP_NAME	varchar (50)	Pentaho sammu (step) nimi, mis sündmuse tekitas	'schema & main tables'

IMP_COLUMNS SKAIS2

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(32)	Lähtesüsteemi kood	'SKAIS2'
SCHEMA_NAME	varchar(128)	Välja/veeru nimi lähtesüsteemis	'SKAIS2'
TABLE_NAME	varchar(128)	Tabeli nimi lähtesüsteemis	'AD_AADDRESS'
COLUMN_NAME	varchar(128)	Välja/veeru nimi andmelaos	'ADS_ADS_OID'
COLUMN_COMMENT	varchar(4000)	Välja/veeru kirjeldus lähtesüsteemis	'Kommentaar'
DATA_TYPE	varchar(128)	Andmetüüp lähtesüsteemis	'VARCHAR2'
DATA_LENGTH	int	Andmevälja pikkus lähtesüsteemis	40
DATA_PRECISION	int	Andmevälja täpsus lähtesüsteemis	18
DATA_SCALE	int	Andmevälja täpsus peale koma lähtesüsteemis	0
NULLABLE	char(1)	Kas puuduvad (NULL) väärtused on lubatud	'Y'
IS_KEY	int	Kas väli/veerg on primaarvõti lähtesüsteemis	0
COLUMN_ID	int	Välja/veeru identifikaator lähtesüsteemis	18

IMP_SCHEMAS SKAIS2

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(32)	Andmeallika/lähtesüsteemi kood. Viide tabelisse DATA_SOURCES	'SKAIS2'
SCHEMA_NAME	varchar(128)	Skeemi nimi lähtesüsteemis	'SKAIS2'

IMP_TABLES SKAIS2

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DATA_SOURCE_CODE	varchar(32)	Andmeallika/lätesüsteemi kood. Viide tabelisse DATA_SOURCES	'SKAIS2'
SCHEMA_NAME	varchar(128)	Skeemi nimi lähtesüsteemis	'SKAIS2'
TABLE_NAME	varchar(128)	Tabeli nimi lähtesüsteemis	'AE_ERIALA'
TABLE_TYPE	varchar(20)	Tabeli tüüp (TABLE - tabel, VIEW - vaade, MVIEW - materialiseeritud vaade)	'TABLE'
TABLE_COMMENT	varchar(4000)	Tabeli kirjeldus lähtesüsteemis	'Kommentaar'

LOAD_LOG SKAIS2

Andmete laadimise logi

Välja nimi	Välja tüüp	Selgitus	Kommentaar	Andmete näide
LOG_ID	int	Logikirje identifikaator. Primaarvõti.		
PROCESS_ID	int	Laadimise protsessi identifikaator.	Iga käivitatud andmete laadimise protsess saab unikaalse identifikaatori.	1
SUBPROCESS_CODE	varchar (50)	Alamprotsessi kood /nimi	Võimalikud väärtused: COMPARE_DATA_MODEL - andmemudeli võrdlemise protsess DMART_LOAD - andmelettide laadimise protsess, DS_LOAD - andmeallika laadimine DWH_LOAD - andmelao laadimine (kogu laadimisprotsess) META_LOAD - metaandmete laadimine ODS_LOAD - ODS tabelite laadimise protsess STG_LOAD - STG tabelite laadimise protsess UPDATE_DATABASE - andmemudeli muutmise protsess	'STG_LOAD'
DATA_SOURCE_CODE	varchar (32)	Andmeallika kood. Viide tabelile ADM_SKAIS2, DATA_SOURCES		'SKAIS2'
RELATED_OBJECT	varchar (50)	Protsessiga seotud objekti (skeemi, andmetabeli vms) kood		'SKAIS2.IS_EEMALOLEK'
START_DT	timestamp	Protsessi käivitamise algusaeg		'2019-06-16 21:29:22'
END_DT	timestamp	Protsessi käivitamise lõppaeg		'2019-06-16 21:29:25'
STATUS	varchar (32)	Laadimise protsessi (lõpetamise) staatus.	Võimalikud väärtused: STARTED - protsessi on alustatud aga see pole veel lõppenud SUCCESS - protsess on lõppenud edukalt WARNING - protsess on lõppenud hoiatusega, konkreetsed hoiatusteaded asuvad tabelis EVENT_LOG ERROR - protsess on lõppenud veaga, konkreetsed veateated asuvad tabelis EVENT_LOG NOT_LOADED - Laadimist ei teostatud, sest eelmine laadimine lõppes veaga.	'ERROR'
IS_IGNORE	int	Kas lõpetamise staatus võib ignoreerida.	Kui eelmine laadimise protsess ei ole lõppenud (on staatuses STARTED) või lõppes veaga (on staatuses ERROR), siis üldjuhul järgmist sama allika laadimisprotsessi ei käivitata. Kui on soov järgmine laadimine ikkagi käivitada, siis tuleb eelmisel laadimisprotsessil märkida IS_IGNORE = 1.	0

PARAMETERS SKAIS2

Süsteemi parameetrid (hetkel ei ole kasutusel).

Välja nimi	Välja tüüp	Selgitus	Andmete näited
PARAMETER_ID	int	Surrogaatvõti	6750
PARAMETER_CODE	varchar(30)	Parameetri kood	'parallel_process_count'
PARAMETER_VALUE	varchar(255)	Parameetri väärtus	'2'

ROW_COUNTS SKAIS2

Kirjete arvu võrdlustabel

Välja nimi	Välja tüüp	Selgitus	Andmete näited
DS_CODE	varchar(32)	Andmeallika/lähtesüsteemi kood	'SKAIS2'
DWH_SCHEMA_NAME	varchar(128)	Andmelao skeemi nimi	'SKAIS2'
DWH_TABLE_NAME	varchar(128)	Andmelao tabeli nimi	'ISIK'
SOURCE_COUNT	int	Kirjete arv lähtesüsteemis	100
DWH_COUNT	int	Kehtivate kirjete arv vastavas andmelao tabelis	100
STATUS_CODE	varchar(32)	Kirjete arvu võrdluse staatus SUCCESS - kirjete arv on võrdne WARNING - kirjete arv on erinev	SUCCESS
STATUS_DESC	varchar(255)	Kirjete arvu täpsem kirjeldus. Võimalikud väärtused correct dwh has less than source dwh has more than source	correct
EVENT_DT	timestamp	Kirjete võrdluse teostamise aeg	'2021-04-17 03:18:05.469'

SCHEMAS SKAIS2

Lähtesüsteemi skeemid

Välja nimi	Välja tüüp	Selgitus	Andmete näited
HEMA_VERSION_ID	int	Skeemi versiooni ID	1
DATA_SOURCE_CODE	varchar(32)	Andmeallika/lähtesüsteemi kood. Viide tabelile DATA_SOURCES	'SKAIS2'
SOURCE_SCHEMA_NAME	varchar(128)	Skeemi nimi lähtesüsteemis	'SKAIS2'
DWH_SCHEMA_NAME	varchar(32)	Skeemi nimi andmelaos	'DW_SKAIS2'
STATUS	varchar(20)	Skeemi andmete laadimise staatus	'LOAD'
IS_ACTIVE	int	Kas skeemi andmete laadimine on aktiivne (võimaldab terve skeemi andmete laadimise lihtsamalt välja lülitada) Hetkel ei kasutata	1
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	1
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2019-06-16 20:36:01'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2019-06-16 20:36:01'
MODIFIED_BY	varchar(32)	Versiooni lisaja/muutja	'pentaho'

TABLES SKAIS2

Tabelite andmed.

Välja nimi	Välja tüüp	Selgitus	Andmete näited
TABLE_VERSION_ID	int	Surrogaatvõti	411
DATA_SOURCE_CODE	varchar (32)	Andmeallika/lähtesüsteemi kood	'SKAIS2'
SOURCE_SCHEMA_NAME	varchar (128)	Skeemi nimi lähtesüsteemis	'SKAIS2'
SOURCE_TABLE_NAME	varchar (128)	Tabeli nimi lähtesüsteemis	'AD_ADDRESS'
DWH_SCHEMA_NAME	varchar (32)	Skeemi nimi andmelaos	'DW_SKAIS2'
DWH_TABLE_NAME	varchar (128)	Tabeli nimi andmelaos	'AD_ADDRESS'
TABLE_TYPE	varchar (20)	Tabeli tüüp (TABLE, VIEW, MVIEW)	'TABLE'
SOURCE_TABLE_COMMENT	varchar (4000)	Tabeli kirjeldus lähtesüsteemis	'Kommentaar'
DWH_TABLE_COMMENT	varchar (4000)	Tabeli kirjeldus andmelaos	'Kommentaar'
LOAD_STATUS	varchar (20)	Laadimise staatus	'LOAD' - Tabeli laadimine on sisse lülitatud 'NEW_LOAD' - Tegemist on uue andmetabeliga (st haldur pole tabeli laadimise staatust üle vaadanud/kinnitanud), tabel on laadimisse sisse lülitatud 'NEW_NOT_LOAD' - Tegemist on uue andmetabeliga, tabel ei ole laadimisse sisse lülitatud 'NOT_LOAD' - tabeli laadimine ei ole sisse lülitatud 'NO_OBJECT' - tabelit ei ole enam lähtesüsteemis (varem oli)
LOAD_METHOD	varchar (20)	Andmete laadimise meetod	'LOAD_MODIFIED' - laaditakse pärast viimast õnnestunud andmete laadimist lisatud /muudetud andmed 'REWRITE' - andmed kirjutatakse üle (st muudatuste ajalugu ei säilitata) 'LOAD_MODIFIED_WOID' - laaditakse pärast viimast õnnestunud andmete laadimist lisatud andmed, tabelil puudub võtmeväli st uued andmed lisatakse, kirjete muutmist ei arvestata (vanade versioonide kehtetuks tunnistamist ei ole).
MODIFIED_FILTER	varchar (255)	Filtritingimus, mida kasutatakse muudetud ja/või lisatud andmete tuvastamiseks (LOAD_MODIFIED) laadimismeetodi korra.	'(case when MUUTM_AEG < DATE '1900-01-01' then DATE '1900-01-01' else MUUTM_AEG end) >=TO_DATE('LAST_LOAD_DT','yyyy-mm-dd hh24:mi:ss)'
LAST_LOAD_EXPIRATION	varchar (255)	Avaldis, mis tuvastab mis aja seisuga andmed on laaditud	'timestampadd('hour',-1, max(MUUTM_AEG))'
DWH_TABLE_STATUS	varchar (30)	Tabeli staatus andmelaos	'NEEDS_CREATION' - tabel vajab loomist 'NEEDS_MODIFICATION' - tabel vajab muutmist 'READY' - tabel on loodud/muudetud
LAST_LOAD_DT	timestamp	Viimase õnnestunud andmete laadimise kuupäev ja kellaaeg	'2019-06-18 14:00:38'
IS_VALID	int	Kas tegemist on kehtiva kirje/versiooniga (0/1)	0
VALID_FROM	timestamp	Kirje/versiooni kehtivuse algusaeg	'2019-06-16 20:36:15'
VALID_TO	timestamp	Kirje/versiooni kehtivuse lõppaeg	'2019-07-08 16:30:17'
MODIFIED_BY	varchar (32)	Versiooni lisaja/muutja	'pentaho'

IS_DM_TABLE	int	Kas tegemist on andmelettide arvutamisel kasutatava andmetabeliga. 0/1 tunnus.	Kui mõne andmelettide laadimises kasutatava tabeli laadimisel tekib viga, siis andmelettide laadimise osa ei käivitata.
IS_ROW_COUNT	int	Kas tabeli laadimisel võrreldakse andmelattu jõudnud kirjete arvu ja lähteallikas olevat kirjete arvu. 0/1 tunnus	1
PRIORITY	int	Andmetabeli laadimise prioriteetsus. Väiksema väärtusega andmetabelid laaditakse enne.	10, 100

Skeemid Skais2

Skeemi nimi	Kirjeldus	Staatus
ADM_SKAIS2	Admin skeem - laadimiste sätted	
ADM_SKAIS2_V2		Vana, pole kasutuses
DW_SKAIS2_DMART	Skais2 Andmelehid	
DW_SKAIS2_ODS	SKAIS2 ODS	
DW_SKAIS2_STG	SKAIS2 STG	
DW_SKAIS2_VALINE_ODS	SKAIS2 VALINE ODS	
DW_SKAIS2_VALINE_STG	SKAIS2 VALINE STG	
DW_SKAIS_BENEFIT_ODS	SKAIS BENEFIT ODS	
DW_SKAIS_BENEFIT_STG	SKAIS BENEFIT STG	
DW_SKAIS_CLIENT_INQUIRY_ODS	SKAIS CLIENT INQUIRY ODS	
DW_SKAIS_CLIENT_INQUIRY_STG	SKAIS CLIENT INQUIRY STG	
DW_SKAIS_DISABILITY_APPLICATION_ODS	SKAIS DISABILITY APPLICATION ODS	
DW_SKAIS_DISABILITY_APPLICATION_STG	SKAIS DISABILITY APPLICATION STG	
DW_SKAIS_DISABILITY_QUESTION_ODS	SKAIS DISABILITY QUESTION ODS	
DW_SKAIS_DISABILITY_QUESTION_STG	SKAIS DISABILITY QUESTION STG	
DW_SKAIS_DOCTOR_ASSESSMENT_ODS	SKAIS DOCTOR ASSESSMENT ODS	
DW_SKAIS_DOCTOR_ASSESSMENT_STG	SKAIS DOCTOR ASSESSMENT STG	
DW_SKAIS_DOCUMENT_MANAGEMENT_ODS	SKAIS DOCUMENT MANAGEMENT ODS	
DW_SKAIS_DOCUMENT_MANAGEMENT_STG	SKAIS DOCUMENT MANAGEMENT STG	
DW_SKAIS_FINANCE_ODS	SKAIS FINANCE ODS	
DW_SKAIS_FINANCE_STG	SKAIS FINANCE STG	
DW_SKAIS_FIN_FACADE_ODS	SKAIS FIN FACADE ODS	
DW_SKAIS_FIN_FACADE_STG	SKAIS FIN FACADE STG	
DW_SKAIS_ISIK_ODS	SKAIS ISIK ODS	
DW_SKAIS_ISIK_STG	SKAIS ISIK STG	
DW_SKAIS_NOTIFICATION_ODS	SKAIS NOTIFICATION ODS	
DW_SKAIS_NOTIFICATION_STG	SKAIS NOTIFICATION STG	
DW_SKAIS_PAKKUMUS_ODS	SKAIS PAKKUMUS ODS	
DW_SKAIS_PAKKUMUS_STG	SKAIS PAKKUMUS STG	
DW_SKAIS_PARENTAL_BENEFIT_ODS	SKAIS PARENTAL BENEFIT ODS	
DW_SKAIS_PARENTAL_BENEFIT_STG	SKAIS PARENTAL BENEFIT STG	
DW_SKAIS_PENSION_SENIORITY_ODS	SKAIS PENSION SENIORITY ODS	
DW_SKAIS_PENSION_SENIORITY_STG	SKAIS PENSION SENIORITY STG	
DW_SKAIS_PEREHYVITIS_ODS	SKAIS PEREHYVITIS ODS	
DW_SKAIS_PEREHYVITIS_STG	SKAIS PEREHYVITIS STG	
DW_SKAIS_PROCEEDING_ODS	SKAIS PROCEEDING ODS	
DW_SKAIS_PROCEEDING_STG	SKAIS PROCEEDING STG	
DW_SKAIS_RFK_CLASSIFIER_ODS	SKAIS RFK CLASSIFIER ODS	

DW_SKAIS_RFK_CLASSIFIER_STG	SKAIS RFK CLASSIFIER STG	
DW_SKAIS_RINAFACADE_ODS	SKAIS RINAFACADE ODS	
DW_SKAIS_RINAFACADE_STG	SKAIS RINAFACADE STG	
DW_SKAIS_TAX_DECLARATION_ODS	SKAIS TAX DECLARATION ODS	
DW_SKAIS_TAX_DECLARATION_STG	SKAIS TAX DECLARATION STG	
DW_SKAIS_TECHNICALAIDXROAD_ODS	SKAIS TECHNICALAIDXROAD ODS	
DW_SKAIS_TECHNICALAIDXROAD_STG	SKAIS TECHNICALAIDXROAD STG	
DW_SKAIS_TECHNICALAID_ODS	SKAIS TECHNICALAID ODS	
DW_SKAIS_TECHNICALAID_STG	SKAIS TECHNICALAID STG	