

Data delivery logging

Document author: anna.hammais@tyks.fi

Automated data delivery processes are currently being developed as Kettle jobs that are stored in the following Git repository:

```
ktpgit.vssh.net:/opt/git/Luovutusprosessit.git
```

Luovutusrekisterin taulujen ohje

Luovutusrekisteriohje

General process description

The development has begun with data to be delivered to Auria biobank. Later added Hercules (Mupet) project where data is delivered to external research environment.

The PostgreSQL schemas used in developing the process are in the *ctp* database:

```
luovutusloki_dev  
auria_dev
```

and the schemas to be used in the production process are:

```
luovutusloki  
auria  
hercules
```

Data delivery in Auria (luovutus) is done through materialized views that are created as selects from existing tables, inner joined with the Auria request SS number list that is valid at the moment when a data delivery round begins.

Data delivery in Hercules is done in principles by the following main steps: 1. SSH Tunnel is created for fetching the "hetu-list" from external research environment (192.168.122.105: Mupet2) 2. "hetu-list" is saved under hercules schema in gradient database. 3. Based on to the source table information in table hercules.upload, the kettle job fetching the data from source tables, creates SSH tunnel and upload the data to (192.168.122.105: Mupet2) by using the SQL function func.get_content. 4. Kettle job also keeps luovutusloki up-to-date during the transformation

More information about Hercules can be found from the Kettle job file: /Luovutusprosessit/hercules/download_update_hercules_data_hercules.kjb

A first draft of the luovutusloki (identical to luovutusloki_dev) database tables is at:

```
ktpanalytics.vssh.net:/opt/shared/KTPDoc/Tietomallit/Autom_luovutusloki/*
```

This is version 2 (2016-05-31):

Kumppani (partner) refers to the data receiver. So far the only receiver of automated data deliveries are Auria and Hercules.

Tapahtumatyyppi (delivery module type) refers to a loggable module in the delivery process. So far, the only ones are "Hetujen haku" (fetching SS numbers from source), "Labrojen luovutus" (lab data delivery) and "Kemokur-tietojen luovutus" (delivery of Kemokur/chemotherapy data).

Luovutuskerta (data delivery process) is a process that (usually) consists of the fetching of SS numbers and the delivery of different data sets for these patients. The fetching, as well as the deliveries, are all listed in the table **Lokitapahtuma** (delivery module).

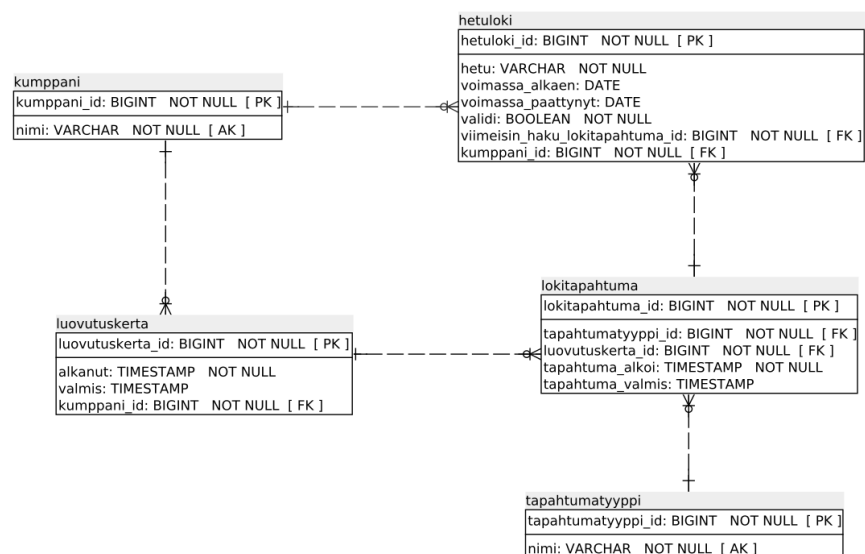


Figure 1: Luovutusloki

The begin and end timestamps of each data delivery process, as well as those of each module, are logged in their respective tables.

The table **Hetuloki** (SSN log) doubles as a SSN list log as well as a listing of currently active SS numbers. Firstly, the SS numbers that have an active consent are marked as `validi=TRUE`. The SS numbers that have withdrawn their consent, i.e. have been on Auria's request list before but have withdrawn their consent (i.e. have disappeared from Auria's list), remain in the Hetuloki but are marked as `valid=FALSE`, and the ending date of the hetu validity is set. For an active hetu, the ending date is null.

A module of type "Hetujen haku" (fetching SS numbers) also updates the Hetuloki (SS number log) table:

1. The list of active SS numbers is retrieved from source.
2. Each SS number that is either not found in Hetuloki or is invalid is input into Hetuloki, with the currently ongoing lokitapahtuma_id set as viimeisin_haku_lokitapahtuma_id. This signifies that the SS number was observed in the Auria list at this time.
3. If an SS number already exists in Hetuloki, then the viimeisin_haku_lokitapahtuma_id is updated to match the currently ongoing one.
4. For each new SS number, the voimassa_alkaen is set as the current date.
5. After the insert/update, those SS numbers that were not seen in the latest list, are marked as invalid and the ending date is set. In this case, the viimeisin_haku_lokitapahtuma_id remains unchanged.

Apart from "Hetujen haku", the other data delivery module types (Tapahtumatyyppi) usually just refresh materialized views, i.e. recreate them with the currently active SS number list.

Which person's data was delivered at which delivery

The schema doesn't specify exactly which data items existed for which patient at the moment of a data delivery. However, it is possible to determine the deliveries and their data types in general that a patient was included in. This can be done by the following query:

```
select kerta.alkanut::date as luovutuskerta_pvm,
```

```

tyyppi.nimi as tapahtuma_nimi,
hetut.voimassa_alkaen as hetu_voimassa_alkaen,
hetut.voimassa_paattynyt as hetu_voimassa_paattynyt
from luovutusloki.luovutuskerta as kerta
-- kaikki kyseisen kumppanin hetut
inner join luovutusloki.hetuloki as hetut on kerta.kumppani_id = hetut.kumppani_id
inner join luovutusloki.lokitapahtuma as tapahtuma on kerta.luovutuskerta_id =
    tapahtuma.luovutuskerta_id
inner join luovutusloki.tapahtumatyyppi as tyyppi on tapahtuma.tapahtumatyyppi_id =
    tyyppi.tapahtumatyyppi_id
where hetut.kumppani_id = (select kumppani_id from luovutusloki.kumppani where nimi = 'Auria')
-- luovutuskerta tapahtunut hetun voimassaolonaikana
and kerta.alkanut between hetut.voimassa_alkaen and coalesce(hetut.voimassa_paattynyt, now())
and hetut.hetu = <some_hetu>
and tyyppi.nimi != 'Hetujen haku';

```

How to use the Kettle jobs

The main job to run is **luovutus.kjb**. It has three parameters:

1. kumppani_nimi (default: Auria)
2. luovutusloki_schema (default: luovutusloki_dev)
3. target_schema (default: auria_dev)

For testing, the defaults are OK, since they refer to the development schemas. For actual data delivery, the parameters should be changed. The schema “luovutusloki” is the actual delivery logging schema, and the kumppani (partner), as well as the partner-specific target_schema should be changed according to the data receiver. Before this, the partner must be added to the luovutusloki.kumppani table, where the column *nimi* is used as the value of this parameter.

Testing

For testing purposes, only a subset of the hetu list can be used. This can be achieved by editing the SQL query in the first step of transformation *fetch_and_insert_hetu_list.ktr*. Adding e.g.

```
"where hetu like '1201%'"
```

allows to fetch a small subset of the list.

Tulevaisuuden kehityssuunnitelmia

Potilastietojen luovutuksista tutkimuksiin ja muihin tarkoituksiin tulee pitää kirjaa. Kliinisen tietopalvelun tietoluovutukset kirjataan kahteen tietokantaskeemaan: luovutusrekisteriin tai luovutuslokiin. Luovutusloki on toistuvien, automatisoitujen tiedonsiirtojen lokitusta varten, ja luovutusrekisteriin kerätään tiedot asiakkaista, tutkimusluvista ja niiden nojalla tehdyistä yksittäisistä tietoluovutuksista. Tietoluovutukset lokitetaan hetukohtaisesti, eli potilaan asiaa tiedustellessa pystytään tarvittaessa tarkistamaan, mihin tutkimuksiin hänen tietojaan on luovutettu, ja onko häntä käsitelty tutkittavana vai kontrollipotilaana.

KTP on itse rakentanut luovutusrekisterin tietokantarakenteen ja java-käyttöliittymän. Käyttöliittymän kautta lisätään tutkimuksia, tutkijoita ja luovutuksia sekä kuvauksia luovutusten tietosisällöstä. Koska KTP:n toiminta on maksullista ja eri projekteihin käytettyä aikaa halutaan seurata, työtuntien seuranta ja projektien hinnat kirjataan myös luovutusrekisteriin.

Tulevaisuudessa luovutusprosessia aiotaan helpottaa automatisoimalla ja lisäämällä luovutusrekisterin käyttöliittymään uusia toiminnallisuuksia. Yhtenä kehitysideana on automaattinen laskutuslistan muodostustoiminto, jolla voidaan valita laskutettaviksi valmiit projektit ja merkitä sitten nämä laskutetuiksi. Tarkoitus on myös automatisoida tilastojen tuotto käyttöliittymän kautta, jolloin olisi entistä helpompaa seurata projektimäärien muutoksia ajassa, eri aineistotyyppien käytön

frekvenssiä ja Tyksin eri toimialueiden aktiivisuutta tietopalvelun käyttäjinä. Näitä asioita on tähän asti tehty SQL-kyselyillä suoraan luovutusrekisterin tietokannasta.

Myös KTP:n projektinhallintaa on toiminnan alkamisen jälkeen kehitetty. Nyt käytössä on yhteinen sähköpostikansio, jossa on alikansiot kullekin asiakkaalle. Käytössä on myös Wekan-tili, jossa kullekin projektille on oma palstansa, jota KTP:n henkilöstö päivittää kun projekti etenee. Projektien dokumentit ja toimitteet tallennetaan myös omiin projektikohtaisiin, yhtenäistä nimeämiskäytäntöä noudattaviin kansioihinsa. Yritysassiakkaille on olemassa yhtenäiset suomen- ja englanninkieliset sopimuspohjat, jolloin sopimusten teko on sujuvaa.