



Huddinge

Systembeskrivning

# Huddingeanalysen

2019-04-23

## Bakgrund

Huddingenanalysen är ett verktyg som gör det enklare att belysa skillnader, likheter och trender mellan olika områden genom att visualisera statistik. Verktöget kan användas för att upptäcka trender och ta fram kunskapsunderlag för att följa upp styrning samt för att analysera och följa upp hållbarhetsarbetet i kommunen. Verktöget är framtaget i samarbete med Sweco under hösten 2018 och våren 2019.

## Visualiseringsverktyget

Visualiseringsverktyget är baserat på ramverket Vizabi som är framtaget av Gapminder Foundation. För mer information om ramverket och kodexempel, se <https://vizabi.org/>.

### Filstruktur

**assets** – Innehåller språkfiler. Här kan man t ex ändra texten som visas på knappar i verktygets gränssnitt.

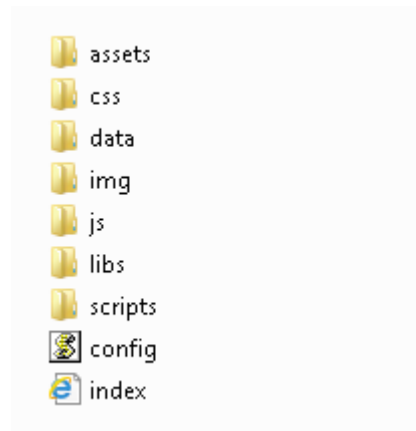
**css** – Innehåller CSS-filer. Om man vill ändra utseendet på komponenter i gränssnittet kan man gå in i filen huddinge.css och göra ändringar.

**data** – Innehåller datasetet som verktyget läser in och visar i verktyget.

**img** – Innehåller bildfilerna som används i verktyget.

**js** – Innehåller de javascript-filer som används i verktyget

**scripts** – Innehåller scripten för export samt create\_database.sql



### Konfigurering av diagram

Konfigurering av bubbeldiagrammet görs i config.js. För exempel på vad man kan konfigurera, se <https://vizabi.org/tutorials/2017/04/02/configure-the-charts/>

### Installation

För att installera visualiseringsverktyget, följ stegen nedan.

1. Installera NodeJS enligt <https://nodejs.org/en/download/>
2. Installera http-server genom att skriva följande i terminalen (cmd)  
npm install http-server -g
3. Installera ddf-validation enligt beskrivning under rubriken "Advanced users can use validator as a command line tool")  
<https://github.com/Gapminder/ddf-validation#datapackage>
4. Packa upp paket1 och paket2 (se bifogade zip-filer) och lägg innehållet i en gemensam mapp 12901041-huddinge-visualisering-segregation
5. Gå in i 12901041-huddinge-visualisering-segregation\scripts och öppna filen export.bat i t ex Notepad++
6. Uppdatera variablerna dirData och dirScript så de matchar enligt där du placerat mappen 12901041-huddinge-visualisering-segregation
7. Starta terminalen (cmd) och gå in i mappen 12901041-huddinge-visualisering-segregation\scripts
8. Skriv call export.bat i terminalen
9. När scriptet kört klart, gå in i mappen 12901041-huddinge-visualisering-segregation via terminalen
10. Skriv http-server i terminalen
11. Surfa in på 127.0.0.1:8080 (eller den adress som visas i terminalen efter att du startat http-server om det är nån annan) i en webbläsare

# Databas – SegregationDW

Databasen har följande struktur

- All ny data läses in i schemat Land
- Transformerat data finns i schemat DW. Allt bearbetat data bor i tabellen dw.transformed\_data förutom ett undantag. Det bearbetade datat för befolkningsundersökningen bor i tabellen dw.population\_survey.
- Bearbetat och aggregerat data finns i schemat DM

## DM

I schemat DM finns det både vyer och tabeller. Data som är mer statiskt, t ex geografiska områden, lagras i tabeller. Data som kommer ändras över tid, dvs data för olika indikatorer, lagras i vyer.

Tabellerna och vyerna som ligger i schemat DM utgör tillsammans ett komplett dataset strukturerat enligt datamodellen DDF. Det data som finns i tabellerna/vyerna och hur tabellerna/vyerna är namngivna följer de riktlinjer som finns i DDF.

### Datamodellen DDF

DDF är ett sätt att organisera data och definiera hur olika delar av data hänger ihop. För en utförlig beskrivning av DDF, [klicka på den här länken för att läsa mer](#).

### Översikt tabeller i DM

#### **ddf—concepts**

Tabellen ddf—concepts är en central tabell för datasetet. I ddf—concepts definieras alla concepts till verket. ddf—concepts måste innehålla alla kolumnnamnen (dvs concepts) som finns i datasetets filer. Om två filer innehåller samma kolumnnamn så räknas det som samma concept.

I ddf—concepts anger man även vilken tag en indikator ska få (se nästa avsnitt för vad en tag är) samt indikatorns beskrivningstext som visas när man klickar på frågetecknet i verket. Det är även här som kartlegendens färgpalett definieras för de olika geografiska områdesindelningarna.

#### **ddf--entities—tag**

I tabellen ddf—entities—tag definieras tags som används för att gruppera indikatorer i menyn som visas när man ska välja en indikator i verket. Vilken tag en indikator har definieras i ddf—concepts.

#### **ddf—entities—nyko3, ddf—entities—nyko5, ddf—entities—municipality\_area, ddf—entities—strategic\_area**

I dessa tabeller definieras de geografiska områdesindelningarna som man kan växla mellan i verket. I kolumnen shape\_lores\_svg definieras de svg-paths som används för kartlegenden i verket.

För att kunna färglägga t ex nyko3 i färger enligt kommundelar så krävs det att det finns en kolumn municipality\_area i tabellen ddf—entities—nyko3 som påvisar hur nyko3-områden relaterar till kommundelar. Det är också den kopplingen som gör att det går att filtrera nyko3-data på kommundelar i verket.

## Översikt vyer i DM (ddf—datapoints)

Beroende på hur många olika sätt man valt att dela upp indikatorn t ex på kön eller geografiskt område så finns det mellan 1-12 vyer per indikator.

### Backup - create\_database.sql

Scriptet create\_database.sql innehåller all SQL som skrivits för att sätta upp tabeller och vyer i DW och DM. Här framgår även hur olika värden räknats fram och aggregerats.

### Export av data

Scriptet export.bat exporterar ett färdigt dataset från databasen i det format som DDF-modellen kräver. **Det är viktigt att filerna i datasetet är av typen .csv, att innehållet är kommaseparerat (ej semikolon) och har encoding UTF-8 (UTF-8 BOM fungerar ej) samt att första raden i csv-filen utgörs av kolumnnamn (concepts).**

För att generera ett dataset med hjälp av export.bat manuellt, följ stegen nedan

1. Gå in i 12901041-huddinge-visualisering-segregation\scripts och öppna filen export.bat i t ex Notepad++
2. Uppdatera variablerna dirData och dirScript så de matchar enligt där du placerat mappen 12901041-huddinge-visualisering-segregation
3. Starta terminalen (cmd) och gå in i mappen 12901041-huddinge-visualisering-segregation\scripts
4. Skriv call export.bat i terminalen

Scriptet export.bat kan schemaläggas om man önskar ett automatiserat dataflöde.

### datapackage.json

För att ett dataset ska vara giltigt så krävs det att det finns en fil datapackage.json i samma mapp som csv-filerna. Datapackage.json fungerar som ett index för datasetet och kan genereras med hjälp av verktyget ddf-validation, se <https://github.com/Gapminder/ddf-validation#datapackage> för instruktioner. **Notera att exportscriptet export.bat genererar datapackage.json automatiskt förutsatt att verktyget ddf-validation är installerat.**

### Hur man lägger till en ny indikator

För exempel se filen create\_database.sql

1. Läs in nytt data i schemat Land
2. Bearbeta och anpassa ditt nya data.
3. Lägg till en eller flera nya kolumner i dw.transformed\_data för ditt nya bearbetade data (antalet nya kolumner beror på hur många du behöver för att kunna beräkna och aggregera indikatorn till olika geografiska nivåer i senare steg)
4. Gör en insert i dw.transformed\_data där du stoppar in värden i kolumnerna area, year samt din nya indikators data i dina nya kolumner.
5. Skapa nya vyer för indikatorn som hämtar data från dw.transformed\_data. Det behövs en vy per geografisk nivå. Om indikatorn delas upp på kön blir det 3 vyer per geografisk nivå. Viktigt att vyerna namnges enligt DDF (se ovan). Kolla i databasen eller create\_database.sql för exempel.
6. Lägg till ett nytt concept (din nya indikators kolumnnamn från vyn) i tabellen dm.ddf—concepts. Observera att om indikatorn är uppdelad på kön så blir det tre nya

concepts att lägga till. Kolla i databasen eller scriptet `create_database.sql` för exempel.

7. Uppdatera scriptet `create_database.sql` med SQL-koden du skrivit för stegen ovan.
8. Exportera ditt nya dataset, följ instruktionerna enligt avsnittet "Export av data" ovan.
9. Validera att allt blivit rätt genom att köra `validate-ddf` i terminalen. Observera att kommandot måste köras inifrån mappen `data/ddf--huddinge`

## Felsökning av dataset

Nedan följer tips på hur man kan felsöka datasetet.

### Felsökning med verktyget `validate-ddf`

- Gå in i mappen `ddf--huddinge` via terminalen
- Om du har verktyget `ddf-validation` installerat (se avsnittet ovan för mer information) så kan du köra `validate-ddf` i terminalen för att validera datasetet. Om du får ett felmeddelande i terminalen – kolla i log-filen som skapas i datasets-mappen för att se vad som är fel i datasetet.

### Vanliga felkällor

- Är filerna i rätt format? Dvs är de UTF-8, har de en första rad med kolumnnamn etc.
- Är filerna korrekt döpta enligt DDF?
- Saknas det ett nytt concept (kolumnnamn) i `ddf--concepts`?
- Har det smyggit sig in ett extra kommatecken någonstans i en text (som inte omges av citationstecken)?