

Rapport datakwaliteit MHL database

Analyse

1. De FK's missen waardoor er geen constraints zitten aan het updaten of verwijderen van id's (of postcodes) waardoor links tussen records tussen tabellen gebroken kunnen worden. Zo kan het ontstaan dat je in je query verwijst naar een kolom in een andere tabel waar hetgeen je naar verwijst niet meer bestaat.

Buiten de data-consistentie maakt het ontbreken van FK's het ook ingewikkelder om complexe relaties tussen tabellen te overzien, wat het correct JOINen van tabellen bemoeilijkt.
2. De omschrijving van de tabellen klopt niet overal. Zo staat er in de tabel mhl_companies bij de omschrijving "Plaatsnamen per gemeente".
3. Hoofdlettergebruik in velden is inconsistent.
4. Bij mhl_cities name is bij sommige waarden een aanduiding van in welk land een stad zich bevindt, terwijl dit te herleiden is via de relatie met commune->district->country en dus redundant is. Hetzelfde geldt voor mhl_communes.
5. In mhl_contacts is er redundancy in het attribuut "contacttype". Zo zijn er bijv. Meerdere records met het contacttype "accountmanager".
6. Namen van tabellen en attributen zijn soms in het Nederlands en soms in het Engels, namen van attributen die fungeren als Foreign Keys zijn niet consistent.
7. In mhl_detaildefs bevat de kolom properties een opsomming van property nummers (e.g. "67,55,17"), wat betekent dat de tabel niet voldoet aan ONV omdat het niet atomic is.
8. Waar in pc_lat_long een postcode een type heeft van char(6), heeft de postcode en p_postcode die in mhl_supplier staat een type van varchar(7). In supplier kan de postcode dus 1 teken langer zijn, waardoor dit voor problemen kan zorgen als er een relatie wordt gelegd tussen de tabellen op basis van postcode. Mogelijk is in het verleden het format geweest "1111 AA" met een spatie ertussen ipv "1111AA", maar het lijkt erop alsof waarden nu allemaal al zonder spatie zijn.

Stappen om kwaliteit te verhogen

1. Al eerder gecreëerde FK's verwijderen. De velden van de FK moeten allemaal terugkomen in de parent tabel. Dit kan bijv. door missende waarden in de child tabel te updaten naar 0 en in de parent tabel een rij op index 0 te maken met alles 0/NULL waarden. Opnieuw FK's toevoegen, bijv. In phpMyAdmin UI of door schrijven van SQL query. Daarbij default constraints van RESTRICT gebruiken.
2. Aanpassen in query of phpMyAdmin.
3. Oude records kunnen niet verholpen worden. Misschien dat het voor de opslag ook niet uitmaakt, enkel als je het gaat tonen op een pagina zou je het kunnen opvangen.
4. ?
5. Nieuwe table maken voor contacttype en naar verwijzen middels FK in mhl_contacts.
6. Zo laten, teveel risico's verbonden aan wijziging.
7. Ik begrijp niet helemaal wat deze entiteit is, maar normaliter zou je bij niet-atomische kolommen dit afsplitsen in een nieuwe tabel waarbij elke waarde een eigen record heeft.
8. Je zou evt. de types gelijk kunnen trekken qua lengte, maar dan wel eerst testen of er records zijn die lengte 7 hebben. Zodat je weet dat er geen tekens verloren gaan als je deze aanpassing maakt. Daarnaast zou je dan bij de plek waar een nieuwe postcode ge-INSERT of ge-UPDATE kan worden een check inbouwen dat deze max. 6 lang is.