Data Kwaliteitsrapport MHL

Een analyse van de data-integriteit van de MHL database

Introductie

De integriteit van de data van MHL kan door intensief gebruik en applicatiewijzigingen niet meer gewaarborgd worden. In dit rapport wordt de data-integriteit beoordeeld op vier maatstaven: *Entity Integrity, Referential Integrity, Domain Integrity* en *Constraint Integrity*. Na iedere bevinding volgen de stappen om de kwaliteit van de data weer op acceptabel niveau te krijgen.

Bevindingen & Adviezen

Entity Integrity

Deze vorm van integriteit betreft de conformiteit naar de derde normaalvorm van Codd [1]. Deze is over het algemeen in orde. Een belangrijk element van de normaalvormen van Codd is echter dat data die afgeleid kan worden geen eigen kolom krijgt. In dit licht is het aan te raden om de `property_DEF` kolom in `mhl_detaildefs` te verwijderen, aangezien deze informatie gesloten zit in `display_name`. In theorie kunnen ook de postcodes in `mhl_suppliers` worden achterhaald aan de hand van de straat en het huisnummer. Dit heeft alleen veel voeten in de aarde omdat een API abonnement moet worden afgesloten en dit moet worden geïmplementeerd. Een ander aspect van de normaalvormen van Codd is dat een cel maar een waarde mag hebben. De `properties` in `mhl_detaildefs` zijn arrays. Daarnaast is hun gebruik niet goed opgenomen, aangezien `propertytype_ID` soms wel is aangegeven maar niet in de array met `properties` staat, en andersom. Aangezien niet iedere detaildefinitie een property heeft, kan met behulp van een extra tabel deze informatie los worden opgeslagen in de database. Tot slot heeft de tabel `mhl_hitcount` geen Primary Key (PK). De huidige PK maakt gebruik van drie waarden. Het loont de moeite om hier ook een `id` kolom toe te voegen met een Auto-Increment functie (AI).

Referential Integrity

Deze vorm van integriteit slaat op de onderlinge relaties tussen tabellen. De Referential Integrity is voor de MHL database **zwaar ondermaats.** Er zijn vrijwel geen enkele Foreign Keys (FK) opgenomen terwijl die er wel moeten zijn. In het Entity Relation Diagram zijn alle nieuwe relaties weergegeven en staat bij iedere kolom of dit een FK is. Een script zal later geleverd worden om deze relaties en *constraints* toe te voegen aan de database. Ook valt op dat de `mhl_brands` tabel geen relatie lijkt te hebben met de rest van de data, waar dit wel zou kunnen qua semantiek.

Domain Integrity

Domain Integrity betreft de correctheid van de datatypen en het uitsluiten van *outliers.* Het gros van de datatypen in de MHL database zijn correct, echter vallen een aantal onregelmatigheden op. Zo zijn `vlevel` in `mhl_detaildefs` en `ulevel` in `mhl_membertypes` een `int(11)` maar hebben beide kolommen alleen waarde 1 of 2. Om opslagruimte te besparen kan dit worden teruggebracht naar een `tinyint`. Ook komen de datatypen van `pc6` in `pc_lat_long` (`char(6)`) niet overeen met de datatypen voor de postcodes in `mhl_suppliers` (`varchar(7)`). Uniformering is dus geboden, zeker omdat ze een relatie met elkaar hebben. Deze relatie is echter niet uitgedrukt met een primary key en foreign key. Tot slot heeft de `content` kolom in `yn_properties` alleen de waarde “Y”. Deze kolom is dus momenteel overbodig.

Constraint Integrity

Deze vorm van integriteit betreft de eisen die aan de data input worden gesteld. De huidige eisen laten bij veel tabellen nog te wensen over. Zo worden op veel plekken lege strings gebruikt als `NULL` waarden. Het is beter om `NULL` waarden in te voegen als een veld optioneel is, zoals bij het `contacttype`, `email` en `tel` in `mhl_contacts`. Hetzelfde geldt voor de `straat`, `huisnummer`, `p_address` en `p_postcode` kolommen in `mhl_suppliers`. Ook missen de `countrycodes` van België, Duitsland, Australië, Denemarken, Oostenrijk en China. Op een aantal plekken komen ook lege strings voor waar onduidelijk is of deze optioneel zijn of niet. Deze zijn: `content` in `mhl_properties`, `display` in `mhl_propertytypes` en `content` in `yn_properties`.

Foreign Key reparatie

Omdat het ontbreken van FKs een dusdanig gemis is, en het repareren hiervan dusdanig ingrijpend is, zullen hier in detail de vervolgstappen uiteen worden gezet, per tabel. In het algemeen zijn er drie opties voor het repareren van deze FKs.

1. De data die wijst naar missende data verwijderen
2. De missende data invullen met een gemiddelde/mediaan/redelijke waarde
3. De missende data invullen met een ‘dummy’ variabele

Hierna zal naar deze opties worden verwezen als “optie 2” of “de tweede optie”, in dit geval het invullen van de missende data met een redelijke waarde.

Contacts

De contacts tabel heeft 67 verwijzingen naar de suppliers tabel zonder referentie. Voor suppliers met ID: 849, 2935, 3178, 3893, 4046, 6991, 7579, 7816, 8549, 8684, 8927, 9598 en 9608 staan een of meerdere contacten genoteerd maar de betreffende supplier is niet meer te vinden in de suppliers tabel. Omdat deze contactgegevens wel waardevol kunnen zijn, is het aan te raden om voor de derde optie te kiezen.

Suppliers

De suppliers tabel heeft in 12 gevallen een membertype van 0, die niet voorkomt in de membertypes tabel. Dit geldt voor de volgende suppliers met ID: 944, 2133, 3874, 5206, 7185, 7248, 7600, 8692, 9239, 9250, 9281 en 9380. Aangezien het in sommige gevallen gaat om serieuze leveranciers (onder andere Philips), is het voor nu verstandig om voor de derde optie te kiezen. Uiteindelijk zal per geval moeten worden nagevraagd wat de status is van het huidige abonnement.

Properties

De properties tabel refereert in 200 gevallen aan niet (meer) bestaande suppliers. Het gaat om de volgende 29 suppliers met ID: 2408, 3284, 4046, 5471, 5705, 5718, 6871, 6886, 6904, 6949, 6991, 7026, 7579, 8553, 8636, 8684, 8805, 8882, 9479, 9511, 9514, 9527, 9535, 9547, 9603, 9604, 9605, 9607 en 9618. De eerste optie lijkt de beste keuze, aangezien er niets bekend is over deze suppliers.
De tabel refereert ook 41 keer aan een propertytype met id 0. Deze bestaat alleen niet in de propertytypes tabel. In 9 gevallen is er bijbehorende content. Het is aan te raden om voor deze rijen een propertytype instance toe te voegen. In de overige 32 gevallen kan redelijkerwijs optie 1 gehanteerd worden.
Verder refereert de ondersteunende yn_properties tabel 28 keer naar een niet bestaande supplier. Het gaat om de volgende 26 suppliers met ID: 495, 724, 726, 908, 1467, 1936, 2408, 2696, 3261, 3893, 4149, 4774, 5471, 5679, 5880, 5888, 8186, 8403, 8553, 8636, 8684, 8988, 9045, en 9046. Aangezien onbekend is om welke suppliers het gaat, is optie 1 een veilige keuze. Dit verhelpt ook het probleem met een missende referentie naar propertytype met id 0.
Tot slot is de detaildefs tabel een zorgenkindje. Door de eerder genoemde arrays als celwaarden is lastig te bepalen wanneer er missende referenties zijn. Wanneer de arrays als losse celwaarden zijn opgenomen in een nieuwe tabel, kan bepaald worden welke propertytype IDs in de detaildefs niet bestaan in de propertytypes tabel.

Hitcounts

De hitcounts tabel refereert in 589 gevallen aan niet (meer) bestaande suppliers. Het gaat om de volgende 87 suppliers met ID: 2, 99, 434, 495, 724, 726, 727, 728, 849, 908, 938, 1228, 1284, 1411, 1467, 1515, 1602, 1658, 1699, 1834, 1858, 1936, 2230, 2337, 2408, 2741, 3433, 3490, 3691, 4096, 4149, 4515, 5033, 5333, 5471, 5606, 5705, 5718, 5719, 5743, 5770, 5851, 5880, 5888, 6666, 6871, 6904, 6949, 6991, 7026, 7128, 7336, 7414, 7416, 7440, 7451, 7479, 7579, 7653, 7661, 7816, 7850, 7862, 8066, 8104, 8186, 8488, 8489, 8553, 8620, 8636, 8684, 8805, 8882, 8927, 8938, 8988, 9034, 9045, 9355, 9421, 9426, 9464, 9467, 9477, 9478 en 9479. De eerste optie lijkt de beste keuze, aangezien er niets bekend is over deze suppliers.

Postcodes Coördinaten

De tabel met postcodes en coördinaten heeft in het slechtste geval voor 1413 postcodes uit de suppliers tabel, niet de hoogte- en breedtegraden. De tweede optie lijkt het beste, concreto het opzoeken van de hoogte- en breedtegraden van iedere missende postcode. Ten eerste moet daarvoor de postcode kolommen (postcode en p_postcode) in de suppliers tabel worden gestandaardiseerd. Dit kan bijvoorbeeld het format “1234AB” zijn. Ten tweede

zou het gebruiken van een API om deze coordinaten te bepalen uitkomst kunnen bieden. Tot slot moeten de referentie naar de pc_lat_long gebruik maken van een id, in plaats van de daadwerkelijk postcode. Dit zal de database versnellen en geeft ook meer ruimte tot het flexibeler opslaan van de postcodes.

Cities & Communes

De tabel met cities heeft in 215 gevallen referenties aan communes die niet zijn opgenomen in de communes tabel. De commune met ID 0 lijkt als een placeholder gebruikt te zijn. Het invullen van de waardes lijkt een goede optie. Ook voor deze taak kan een API in gebruik worden genomen.

Rubrieken

De rubrieken-suppliers pivot tabel refereert in 621 gevallen aan suppliers zonder dat deze supplier bestaat in de suppliers tabel. Dit zorgt voor moeizame boekhouding. Het is dus aan te raden om deze referenties te verwijderen. Verder refereert deze pivot tabel in 443 gevallen aan een niet bestaande rubriek. Ook hiervoor is het aan te raden om optie 1 te hanteren.

Conclusie

De MHL database bevindt zich in slechte staat. Op ieder gebied van data-integriteit zijn fouten te bespeuren met als grootste defect het ontbreken van FKs. Het bijgeleverde script kan hier orde op zaken stellen. Het is ten harte aan te bevelen om de overige gesuggereerde wijzigingen in dit document over te nemen.

Bronnen

[1] Codd, E. F. "Further Normalization of the Data Base Relational Model". (Presented at Courant Computer Science Symposia Series 6, "Data Base Systems", New York City, May 24–25, 1971.) IBM Research Report RJ909 (August 31, 1971). Republished in Randall J. Rustin (ed.), *Data Base Systems: Courant Computer Science Symposia Series 6*. Prentice-Hall, 1972.