**Insights using MySQL to explore Global Trade Item Number (GTIN) data structures**

Randy Lisbona, Marvin Scott, Vinh Le

**Abstract**

GS1 is a global organization that maintains standards for numbering systems used between trading partners. GTIN (Global Trade Item Number) is a family of product ID codes used worldwide between trading partners to identify products via linear or 2D barcodes on the product or packaging. Commonly referred to as UPC codes in the USA. This project will provide insight on the format and usage of GTIN product codes.

An open source GTIN subset > 100K records will be downloaded and installed on a local mySQL database instance. Open source documentation will be reviewed to better understand the GTIN data. A graphical schema of the database will be created with example data from each table to visualize the relationship between tables. Exploratory queries will be created to summarize selected fields such as brand, and packaging level. A summary list of mySQL commands used in the analysis will be provided along with impressions on ease of use, intuitiveness, and effectiveness.

Figure 1 - GTIN formats



# GS1.org History

GS1.org is the global standards organization that manages a system of unique identification numbers used between trading partners in over 150 countries worldwide. GS1 traces its roots to the Uniform Product Code Council (UCC) which was established in the USA in 1973 to manage UPC barcodes used in North America, and to the European Article Numbering Association (EAN International) established in 1977 to develop a compatible barcode identification system outside of North America. GS1 was launched in 2005 to combine the two standards organizations into one international organization. [1]

# GTIN formats

GTIN (Global Trade Item Number) is just one of the numbering systems managed by GS1:

* **Global Trade Item Number (GTIN).**
* Global Location Number (GLN).
* Serial Shipping Container Code (SSCC).
* Global Returnable Asset Identifier (GRAI).
* Global Individual Asset Identifier (GIAI).
* Global Service Relation Number (GSRN).
* Global Document Type Identifier (GDTI).
* Global Shipment Identification Number (GSIN).
* Global Item Number for Consignment (GINC).
* Global Coupon Number (GCN).
* Component / Part Identifier (CPID).

A GTIN consists of four parts, an optional Application Identifier, the Company Prefix, the Item Reference, and a check digit. Notice that price is not in the GTIN, the GTIN is used to lookup product price in an external database. See Figure 1. [1]

* GTIN-8 is a truncated 8 digit GSI identification key used on packages with limited label space, such as chewing gum.
* GTIN-12 is 12 digit GSI identification key consisting of a U.P.C. company prefix, item reference, and check digit.
* GTIN-13 is a 13 digit GSI identification key consisting of a GS1 Company prefix, item reference, and check digit
* GTIN-14 is a 14 digit GSI identification key consisting of an indicator digit (1-9), GS1 company prefix, item key, and check digit.

## Indicator Prefix



Table 2.1 - Figure 3 Decoding

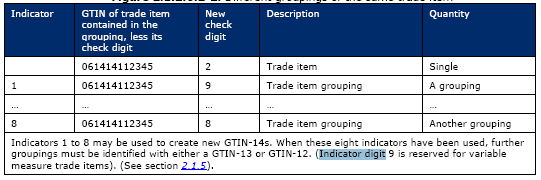


Figure 2 - Indicators

The Indicator Prefix Digit is used to define a grouping, or packaging level. For instance, in Figure 2, [1] we see an example GTIN of 061414112345, let’s say this was a 2oz bag of candy corn. Adding an indicator prefix of “1” could denote a bulk pack of 12 2oz packages, indicator prefix “2” could be used for a 24 pack, indicator prefix “3” could be a “Gross” pack (144) of 2oz candy corn packages, and so on. If more than 8 packaging levels are needed, then a new GTIN-12 or GTIN-13 is required.

## Variable Measure Trade Items

Indicator prefix “9” is reserved for variable measure trade items. Variable measure items are trade items that cannot guarantee consistent weight, size, or length due to the production process (e.g. meat, bulk cheese).

## Data Carrier

A data carrier is a means of representing GS1 identification codes in a machine readable form. 1D Linear barcodes such as UPC-A or EAN-13 are widely used. Matrix symbols such as QR code and Data Matrix are also supported, as well as RFID tags.

## Examples

Figure 3 shows an example barcode from a box of chocolates. The format is GTIN-13 and the data carrier is an EAN-13 Linear barcode.



Figure 3 - EAN-13 barcode symbol

Table 2.1 shows decoded values from figure 3. The first three digits (400) indicate the country that issued the Global Company Prefix. In this case it was Germany. The country that issued the GCP is not necessarily the home country of the company. For instance, a US based company may get a GCP issued in a foreign country. The Prefix plus the company code 81555 results in a unique company code worldwide (40081555).

The item code is 02020, Companies assign their own item codes. The calculated check digit is 7. Check digit calculations are outlined in section 7.9 of the GS1\_General\_Specifications document. [1]

Figure 4 is a chocolate candy bar from the same company, note the item code of 03300 and check digit 9.

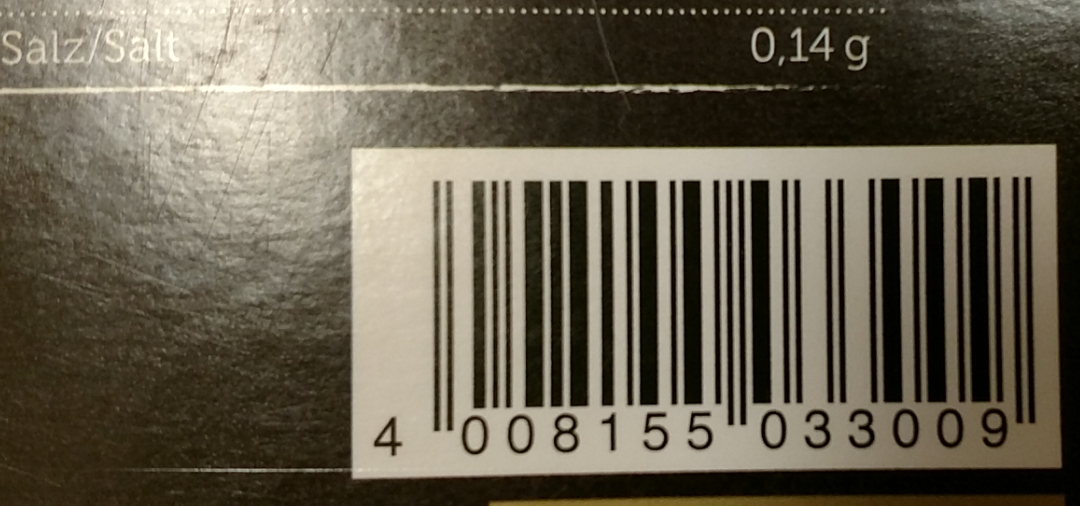


Figure 4- EAN-13 barcode symbol 2

Figure 5 shows a GTIN-12 encoded in a UPC-A linear barcode. 8 indicates the GPC was assigned in the USA, 84912 is the GPC for Post Consumer Brands, LLC. The Item number is 00471 and it was from a 581-gram box of Grape Nuts Cereal.



Figure 5 - UPC-A barcode symbol

# Product Open Data

GS1 does not maintain a worldwide list of Global Trade Item numbers (GTIN), GS1 simply assigns Global Company Code prefixes to ensure unique GTIN values. Each company using GTIN maintains their own list of item numbers and shares this with their trading partners. There are several initiatives to create a global open source GTIN list. For this project the Product Open Data (POD) database was used. This is far from a complete database in its current state, but does demonstrate the type of data linked to GTIN’s.

## Download and install POD

The POD database was downloaded and installed locally in MySQL. [2] The Specification file was also downloaded. [3] Queries were written to pull sample records from each table.

## POD Observations

Thirteen tables containing 2.6 million rows were found in the POD database (see figure 6). Example records were extracted from each table and key fields were examined to generate a schema. See Appendix B, Figure B1. Discrepancies from the schema in the specification file found with the POD database were resolved using key field names and matching records in the POD database. While there were at least a few records in each table, matching records in related tables proved to be disappointing. See Appendix C, figure C1 for a summary of matching records in the 7 largest tables. Of the 2.6 million original rows, only 52 were linked between 6 tables, 0 for 7 or more tables. The website seems to imply POD is an attempt at a global GTIN registry but it appears it may have been abandoned in early 2014, the last revision date on the database dump.

Figure 6 - Record counts and key fields in POD database



## Metadata

Figure 7 shows average percentage of some of the metadata collected for the main objects of this exercise. Percent amount for the Table Rows, Average Row Length, Data Length and Index Length for each main object are illustrated below.

Figure 7 - Object Metadata



|  |
| --- |
| Figure 8 - Global Location Names |
|  |

## Global Location Name

Figure 8 displays distinct count of the GTINs and their associated Global Location Names for several example companies found in the POD database.GTIN (Global Trade Item Number) is the number you most commonly see printed as part of a product’s bar code. The number identifies any item traded in the global supply chain that can be priced, ordered or invoiced.

## Global Location Number

Global Location Number (GLN) is a type of GS1 identification key used to determine any location that needs to be identified in the supply chain. It is designed to efficiently identify physical locations, Operational locations, and Party. GLN are used for retrieving information from different entities such as hospitals, medical supplies, delivery point, warehouses, and banks. It is constructed with a13-digit numeric structure with a prefix, location reference, and check digit. GS1 Company Prefix is assigned by a GS1 Member Organization a user. Location reference is allocated by the company to a specific location. Check digit is calculated according to a standard algorithm, which helps to ensure integrity. GLNs provides companies with a method of identifying location within and outside their company to prevent duplication, complexity, and significance problems. The benefit of using GLNs is that it is unique, multi sectoral, and international.

A Global Location Number provides a unique and unambiguous identification of physical locations and parties used in the supply chain or participating in a business process.



# Summary

Without Global Company Prefixes and Global Trade item numbers, it would be possible and highly likely that different companies could choose to use the same identification number (Part number) on their products. It is hard to imagine how global trade would work without Global Trade Item Numbers and the GS1 organization in place to ensure a unique numbering system worldwide. GS1 does not maintain a list of global trade item numbers, they just maintain the Global Company Prefixes used to ensure unique item numbers. The Product Open Database (POD) gave an interesting peak into the detailed data related to GTIN, but it’s small size and fractured nature were disappointing.

There are several other “Open” implementations of Global Trade Item numbers. See Appendix A, table A2. POD first appeared to be a complete subset but after further investigation it was found to be missing enough linked records in related tables to be very useful.

# References

|  |  |
| --- | --- |
| [1] | GS1.org, "The global Language of Business," 11 2016. [Online]. Available: http://www.gs1.org/gs1-source/latest GS1\_General\_Specifications.pdf. |
| [2] | "Product Open Data," 01 01 2014. [Online]. Available: http://www.product-open-data.com/download/ "POD Database - Dump". [Accessed 11 2016]. |
| [3] | "Product Open Data," 13 11 2013. [Online]. Available: http://www.product-open-data.com/download/ "POD Database - Specifications". [Accessed 11 2016]. |

**Appendix A Data sources and related works**

Github Repository : <https://github.com/rlisbona/MSDS-7330-Term-Paper-1>

The Github Repository contains files used in this project.

Table A1 shows links to data sources downloaded for this project

Table A1 - Data Sources

|  |  |
| --- | --- |
| Product Open Data – Subset of GTIN | <http://www.product-open-data.com/en/1-home.html> |
| POD database SQL Create and Load | <http://www.product-open-data.com/docs/pod_web_2014.01.01_01.sql.gz> |
| POD database Specification | <http://www.product-open-data.com/docs/POD-SPECS-2013.11.13_01.xlsx> |

Table A2 shows websites similar to Product Open Data. Most of these limit searches to just a few GTIN records at a time and do not have a database to download.

Table A2 -- Related Works

|  |  |
| --- | --- |
| Open EAN/GTIN Database | http://opengtindb.org/ |
| Outpan | https://www.outpan.com/ |
| Datakick | https://www.datakick.org/ |
| EAN-Search | http://www.ean-search.org/ |

**Appendix B POD database Schema**

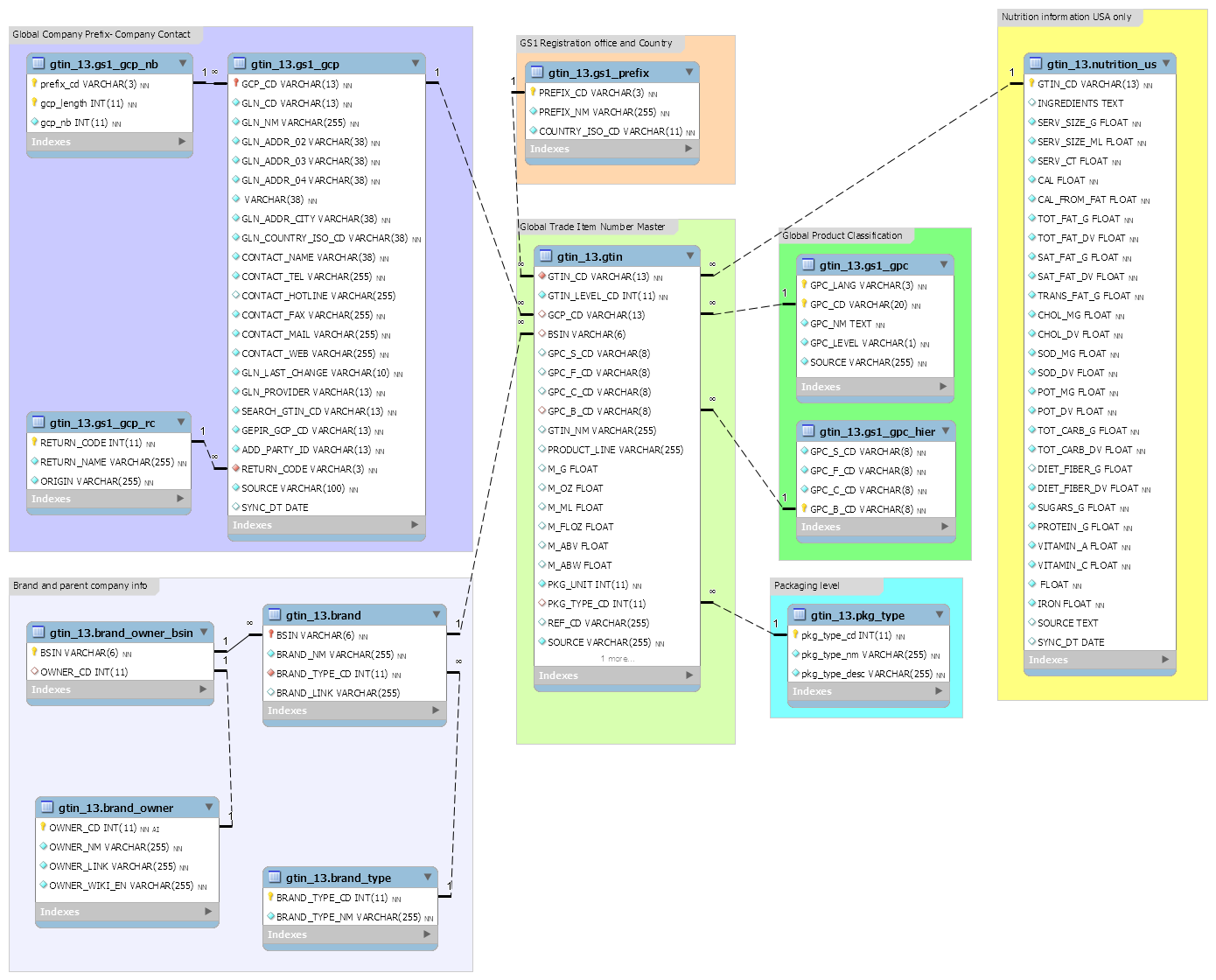


Figure B1 - POD database Schema

**Appendix C – MySql Exploratory code**

Figure C1 – Exploratory Joins

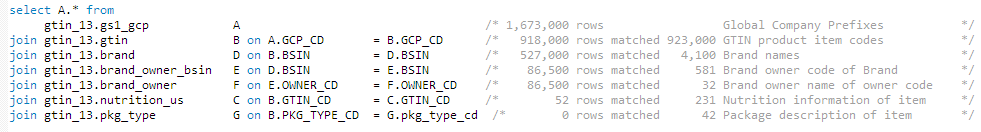


Figure C1 shows exploratory join record counts. As more tables were added to the join, fewer and fewer rows matched between all tables. The conclusion from this query is that the example data in the POD GTIN database is not a complete subset of GTIN.

**MySQL queries used to explore and create charts**

/\* Explore GTIN\_13 tables - Randy Lisbona 10/29/2016 \*/

/\* Database storage calculation using example from www.a2hosting.com\*/

/\* https://www.a2hosting.com/kb/developer-corner/mysql/determining-the-size-of-mysql-databases-and-tables \*/

SELECT table\_schema AS "Database",

ROUND(SUM(data\_length + index\_length) / 1024 / 1024, 2) AS "Size in (MB)"

FROM information\_schema.TABLES

WHERE table\_schema = "gtin\_13";

SHOW VARIABLES LIKE "%version%";

select version();

/\* count columns per table \*/

SELECT table\_name, COUNT(\*) totalColumns

FROM INFORMATION\_SCHEMA.COLUMNS

WHERE table\_schema = "gtin\_13"

group by table\_name;

/\* this is giving a slightly different record count for some tables, not sure why, use individual queries instead

select TABLE\_NAME, TABLE\_ROWS from information\_schema.tables where information\_schema.tables.table\_schema = 'gtin\_13';

/\* TABLE\_NAME TABLE\_ROWS \*/

/\* brand 4151 \*/

/\* brand\_group 3 \*/

/\* brand\_owner 32 \*/

/\* brand\_owner\_bsin 581 \*/

/\* brand\_type 2 \*/

/\* gs1\_gcp 1549550 \*/

/\* gs1\_gcp\_nb 264 \*/

/\* gs1\_gcp\_rc 16 \*/

/\* gs1\_gpc 38760 \*/

/\* gs1\_gpc\_hier 3298 \*/

/\* gs1\_prefix 1000 \*/

/\* gtin 844270 \*/

/\* label 2 \*/

/\* label\_gtin 3 \*/

/\* nutrition\_us 231 \*/

/\* pkg\_type 42 \*/

select \* from information\_schema.REFERENTIAL\_CONSTRAINTS where information\_schema.REFERENTIAL\_CONSTRAINTS.constraint\_schema = 'gtin\_13';

/\* no referential constraints found \*/

select \* from information\_schema.TABLE\_CONSTRAINTS where information\_schema.TABLE\_CONSTRAINTS.constraint\_schema = 'gtin\_13';

/\* All tables have Primary Keys \*/

select \* from information\_schema.KEY\_COLUMN\_USAGE where information\_schema.KEY\_COLUMN\_USAGE.constraint\_schema = 'gtin\_13';

/\* Tables and Keys

TABLE\_NAME Key1 Key2

brand BSIN

brand\_group BSIN

brand\_owner OWNER\_CD

brand\_owner\_bsin BSIN

brand\_type BRAND\_TYPE\_CD

gs1\_gcp GCP\_CD

gs1\_gcp\_nb prefix\_cd gcp\_length

gs1\_gcp\_rc RETURN\_CODE

gs1\_gpc GPC\_LANG GPC\_CD

gs1\_gpc\_hier GPC\_B\_CD

gs1\_prefix PREFIX\_CD

gtin GTIN\_CD

label LABEL\_ID

nutrition\_us GTIN\_CD

pkg\_type pkg\_type\_cd

\*/

/\* select record counts from all tables \*/

select 'brand' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.brand union

select 'brand\_owner' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.brand\_owner union

select 'brand\_owner\_bsin' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.brand\_owner\_bsin union

select 'brand\_type' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.brand\_type union

select 'gs1\_gcp' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gs1\_gcp union

select 'gs1\_gcp\_nb' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gs1\_gcp\_nb union

select 'gs1\_gcp\_rc' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gs1\_gcp\_rc union

select 'gs1\_gpc' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gs1\_gpc union

select 'gs1\_gpc\_hier' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gs1\_gpc\_hier union

select 'gs1\_prefix' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gs1\_prefix union

select 'gtin' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.gtin union

select 'nutrition\_us' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.nutrition\_us union

select 'pkg\_type' as 'TABLE\_NAME', count(\*) as 'TABLE\_ROWS' from gtin\_13.pkg\_type;

/\* Individual tables \*/

drop view if exists brand\_owner\_bsin\_subset;

create view brand\_owner\_bsin\_subset as

SELECT \* FROM gtin\_13.brand\_owner\_bsin

where OWNER\_CD between 27 and 30

order by OWNER\_CD, BSIN;

SELECT \* FROM gtin\_13.brand\_type;

SELECT \* FROM gtin\_13.pkg\_type;

select distinct(pkg\_type\_cd) from gtin; /\* Check how many package types there are: 23 rows returned \*/

select distinct(GCP\_CD) from gtin; /\* 52918 distinct rows returned for Company code \*/

/\* 15,502 distinct brand single identification number (BSIN) matched in gtin \*/

select count(\*) as Freq, A.GCP\_CD, A.BSIN, B.BRAND\_NM from gtin\_13.gtin A join gtin\_13.brand B on a.BSIN = B.BSIN GROUP BY A.GCP\_CD, A.BSIN, B.BRAND\_NM ;

select A.GTIN\_CD, A.GCP\_CD, A.BSIN, B.BRAND\_NM

from gtin\_13.gtin A left join gtin\_13.brand B on a.BSIN = B.BSIN

GROUP BY A.GCP\_CD, A.BSIN, B.BRAND\_NM

order by A.GTIN\_CD ; /\* 62,849 of 922,000 gtin records have BSIN codes \*/

select distinct(length(GTIN\_CD)) from gtin; /\* check this syntax \*/

select GTIN\_CD, length(GTIN\_CD) as GTIN\_LEN, GCP\_CD, length(GCP\_CD) as GCP\_LEN from gtin order by length(GTIN\_CD);

select A.GPC\_NM, B.GTIN\_CD, B.GPC\_S\_CD, B.GPC\_C\_CD, B.GPC\_C\_CD, B.PRODUCT\_LINE from

gtin\_13.gs1\_gpc A /\* 1,673,000 rows returned -Product names in various languages \*/

join gtin\_13.gtin B on B.GPC\_B\_CD = A.GPC\_CD ; /\*and A.GPC\_LANG = "EN";\*/

select M\_G, M\_OZ, count(\*) from gtin\_13.gtin

group by M\_G, M\_OZ;

drop view if exists gtin\_subset;

create view gtin\_subset as

SELECT A.\* FROM gtin\_13.gtin A join gtin\_13.nutrition\_us B on

A.GTIN\_CD = B.GTIN\_CD

where b.INGREDIENTS REGEXP '.WHEAT.'

limit 20;

drop view if exists gs1\_gcp\_subset;

create view gs1\_gcp\_subset as

SELECT A.\* FROM gtin\_13.gs1\_gcp A left join gtin\_subset B on

A.GCP\_CD = B.GCP\_CD or a.GCP\_CD = '73410'

order by GCP\_CD

limit 20;

/\* find records with 73410 anywhere in the GCP\_CD field \*/

SELECT A.\* FROM gtin\_13.gs1\_gcp A

where A.GCP\_CD REGEXP '.73410.';

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* The deeper we go (linking more tables) the fewer matching records we find \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Recommend using primarily the gtin table and brand table, gts\_gcp doesn't have much useful information \*\*\*\*\*\*\*\*\*\*/

select A.\* from

gtin\_13.gs1\_gcp A /\* 1,673,000 rows Global Company Prefixes \*/

join gtin\_13.gtin B on A.GCP\_CD = B.GCP\_CD /\* 918,000 rows matched 923,000 GTIN product item codes \*/

join gtin\_13.brand D on B.BSIN = D.BSIN /\* 527,000 rows matched 4,100 Brand names \*/

join gtin\_13.brand\_owner\_bsin E on D.BSIN = E.BSIN /\* 86,500 rows matched 581 Brand owner code of Brand \*/

join gtin\_13.brand\_owner F on E.OWNER\_CD = F.OWNER\_CD /\* 86,500 rows matched 32 Brand owner name of owner code \*/

join gtin\_13.nutrition\_us C on B.GTIN\_CD = C.GTIN\_CD /\* 52 rows matched 231 Nutrition information of item \*/

join gtin\_13.pkg\_type G on B.PKG\_TYPE\_CD = G.pkg\_type\_cd /\* 0 rows matched 42 Package description of item \*/

/\* extract example rows from each table \*/

SELECT \* FROM gtin\_13.gs1\_gcp\_nb

where prefix\_cd between '800' and '900';

SELECT \* FROM gtin\_13.gs1\_gcp\_rc;

SELECT \* FROM gtin\_13.gs1\_gpc

where GPC\_LANG = 'EN' ;

limit 15;

SELECT \* FROM gtin\_13.gs1\_gpc\_hier

limit 15;

SELECT \* FROM gtin\_13.gs1\_prefix

where COUNTRY\_ISO\_CD ='US'

limit 10;

SELECT A.\* FROM gtin\_13.nutrition\_us A join gtin\_subset B on

A.GTIN\_CD = B.GTIN\_CD

limit 200;

SELECT A.\* FROM gtin\_13.nutrition\_us A join gtin B on

A.GTIN\_CD = B.GTIN\_CD

limit 20;

SELECT \* FROM gtin\_13.label\_gtin;

SELECT \* FROM gtin\_13.label;

SELECT A.\* FROM gtin\_13.brand as A join brand\_owner\_bsin\_subset as B on

A.BSIN = B.BSIN

order by A.BSIN;

SELECT \* FROM gtin\_13.brand\_group;

SELECT \* FROM gtin\_13.brand\_owner;

select Count(\*) from gtin\_13.gtin where BSIN is null;

select count(\*), length(trim(leading '0' from GTIN\_CD)) as length from gtin\_13.gtin group by length(trim(leading '0' from GTIN\_CD));

/\* count number of rows by brand URL \*/

select count(\*) , BRAND\_LINK from gtin\_13.brand group BY BRAND\_LINK order by count(\*) DESC;

/\* count number of rows where the Global Location Number name is null or '', these are just showing the country that issued the code 512,813 rows \*/

select count(\*), ifnull(GLN\_CD,'') as GLN\_CD, GLN\_NM from gtin\_13.gs1\_gcp group by ifnull(GLN\_CD,''), GLN\_NM order by count(\*) desc;

/\* count the records that have a Gloal Location Number Code 505,740 \*/

select count(\*), GLN\_CD from gtin\_13.gs1\_gcp group by GLN\_CD having not isnull(GLN\_CD) order by count(\*) desc;

/\* Data for chart of Global Location Numbers\*/

select GCP\_CD, ifnull(GLN\_CD,'') as GLN\_CD, GLN\_NM from gtin\_13.gs1\_gcp

where is null(GLN\_NM)

order by ifnull(GLN\_CD,''), GCP\_CD ;

/\*Returns 1,095,279 rows with '' Global Location numbers out of 1,549,550\*/

select GCP\_CD, GLN\_CD, GLN\_NM from gtin\_13.gs1\_gcp

where GLN\_CD = ''

order by GLN\_CD ;

/\* Returns 540580 Global Location Numbers with company codes out of 1,549,550\*/

select GCP\_CD, GLN\_CD, GLN\_NM from gtin\_13.gs1\_gcp

where GLN\_CD <> '' and not (GLN\_NM like '%GS1%' or GLN\_NM like '%Unknown country%' or GLN\_NM = '' or GLN\_NM like 'Prefix never allocated%' or GLN\_NM like 'ReturnCode%');

order by GLN\_NM ;

/\* Returns 35,066 rows that have a Global Location Number but only the country that issued it, no company \*/

select GCP\_CD, GLN\_CD, GLN\_NM from gtin\_13.gs1\_gcp

where GLN\_CD <> '' and GLN\_NM like '%GS1%';

order by GLN\_CD ;

/\* Returns 2739 rows with an garbage GLN\_NM \*/

select GCP\_CD, GLN\_CD, GLN\_NM from gtin\_13.gs1\_gcp

where GLN\_CD <> '' and ( GLN\_NM like '%Unknown country%' or GLN\_NM = '' or GLN\_NM like 'Prefix never allocated%' or GLN\_NM like 'ReturnCode%')

order by GLN\_CD ;

/\* Overall summary table \*/

select Status, count(\*) from (

select GLN\_CD, GLN\_NM,

case

when GLN\_NM like '%GS1%' then 'GS1 Issue Country only'

when GLN\_CD = '' then 'Invalid or Missing'

when GLN\_NM like '%Unknown country%' or GLN\_NM = '' or GLN\_NM like 'Prefix never allocated%' or GLN\_NM like 'ReturnCode%' then 'Invalid or Missing'

else 'Global Location Number found'

end as Status

from gtin\_13.gs1\_gcp ) as derivedtable

group by Status

order by count(\*) desc;

/\* end Randy Lisbona exporatory queries \*/