SQL Architecture

# Introduction

The SQL database for OpenStore uses a single SQL database table to store all data.

NBrightBuy

It has 2 linked tables that are used for indexing and langue, which helps OpenStore have quick database retrieval.

NBrightBuyIdx

NBrightBuyLang

These 2 tables are data duplicated of NBrightBuy, purely to help with database speed and are maintained by using triggers on the main data table (NBrightBuy).

The data structure works as a NoSQL database, but has the advantage that SQL can be used on the data. All data is saved in the SQL Server standard XML.

We will refer to these 3 tables as the OpenStore Database.

# NBrightBuy

The NBrightBuy table is a generic structure that uses XML to store data. The other columns indicate relationship between records or external sources. Different columns will be used depending on the data record TypeCode.

|  |  |
| --- | --- |
| Column Name | Description |
| ItemId | Identity column, unique SQL record id. |
| PortalId | PortalId in DNN |
| ModuleId | ModuleId in DNN |
| TypeCode | The data record type, to identify what data is kept in the record XML |
| XMLData | XML data source |
| GUIDKey | Identifying ID. Despite the name this is NOT guaranteed to be unique and is not usually a GUID. |
| ModifiedDate | Date last changed |
| TextData | Text data, not used in OpenStore |
| XrefItemId | Cross reference id, that links to another record in the table. |
| ParentItemId | Parent reference id, that links to another record in the table. |
| Lang | Identifies a localized record. |
| UserId | User id |
| LegacyItemId | Legacy id, often used for import of data from other systems. |

# OpenStore Database TypeCode

The TypeCode column in the NBrightBuy table is the key identifier for data. You can think of this column as the equivalent of a table name. Each TypeCode in the table holds a different type of data.

|  |  |
| --- | --- |
| TypeCode | Description |
| CART | Cart Data |
| CATCASCADE | Products in all sub-categories |
| CATEGORY | Category & Property data |
| CATEGORYLANG | Category & Property localized data |
| CATGRPXREF | Category Filter Groups |
| CATXREF | Which Products are in which category. |
| CLIENT | Client data, favourite lists |
| DISCOUNTCODE | Discount code data |
| DISCOUNTCODELANG | Discount code localized data |
| GROUP | Property groups data |
| GROUPLANG | Property groups localized data. |
| ORDER | Order data |
| PLUGIN | Plugin data |
| PRD | Product data |
| PRDLANG | Product localized data |
| PRDXREF | Product related products |
| SETTINGS | Module and store settings |
| SHIPPING | Shipping data |
| USERDATA | User profile and addresses. |

There will also be extra TypeCode from other plugins to the system. When creating a plugin we MUST NOT use any TypeCode that is already used.

# Localized Data

In OpenStore we need to keep data for multiple languages, this is done by creating a localized or “\*LANG” record in the data. For example, for products you have 2 record types, PRD & PRDLANG. The PRDLANG record contains the localized data, with the “Lang” column populated with the localized culturecode.

# XML Data

The xml data in the “XMLData” column is the core data for OpenStore. The root node for the xml is “<genxml>”.

For localized records we use the “LANG” suffix, the return data will use the “NBrightBuyLang.XMLData” column. This column is a merge of the standard xml record, with the localized xml inserted into a node “<lang>”

<genxml>

[Normal Data]

<lang>

<genxml>

[Localized Data]

</genxml>

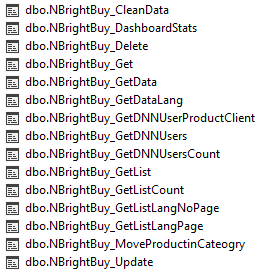
</lang>

</genxml>

# Triggers

Trigger\_NBrightBuyIdx If the only trigger, which is linked to the NBrightBuy table. When the NBrightBuy table is updated, the trigger updates the NBrightBuyIdx and NBrightBuyLang tables, to keep them in-line with the NBrightBuy data table.

# Store Procedures

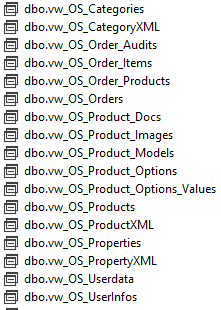


These Stored Procedures are used by OpenStore to maintain the NBrightBuy table and no other Stored Procedures are used in any database access.

|  |  |
| --- | --- |
| Name | Description |
| NBrightBuy\_CleanData | Remove all data that is linked to a deleted module. |
| NBrightBuy\_DashboardStats | Get all data required to display the OpenStore Admin dashboard. |
| NBrightBuy\_Delete | Delete records |
| NBrightBuy\_Get | Get record by id. WITH localized data |
| NBrightBuy\_GetData | Get record by id, WITHOUT localized data |
| NBrightBuy\_GetDataLang | Get localized record. |
| NBrightBuy\_GetDNNUserProductClient | Get DNN user data. |
| NBrightBuy\_GetDNNUsers | Get DNN user data list. |
| NBrightBuy\_GetDNNUsersCount | Get DNN User Count. |
| NBrightBuy\_GetList | Get data list. (Calls “NBrightBuy\_GetListLangNoPage” or “NBrightBuy\_GetListLangPage” depending on parameters) |
| NBrightBuy\_GetListCount | Get data list count. |
| NBrightBuy\_GetListLangNoPage | Get data list without paging. |
| NBrightBuy\_GetListLangPage | Get data list with paging. |
| NBrightBuy\_MoveProductinCateogry | Moves product sort order in a category. |
| NBrightBuy\_Update | Update data record. |

# OpenStore SQL Views

In v4.1 a number of SQL views have been created to make SQL reports easier. Because the data is in XML, SQL Server does not support view updates, therefore you’ll not be able to update through these tables.



# SQL examples

## Get data for a data type (example PRD for products)

SELECT \* FROM [dbo].[NBrightBuy] where typecode = 'PRD'

## Return all data linked to 1 product

SELECT \* FROM [dbo].[NBrightBuy] where itemid = 83 or ParentItemId = 83 or XrefItemId = 83

## Delete ALL products in DB

delete from NBrightBuy where typecode = 'PRD'

delete from NBrightBuy where typecode = 'PRDLANG'

delete from NBrightBuy where typecode = 'PRDXREF'

delete from NBrightBuy where typecode = 'USERPRDXREF'

delete from NBrightBuy where typecode = 'CATCASCADE'

delete from NBrightBuy where typecode = 'CATXREF'

## Delete ALL Categories in DB

delete from NBrightBuy where typecode = 'CATEGORY' and [XMLData].value('(genxml/dropdownlist/ddlgrouptype)[1]','nvarchar(max)') = 'cat'

delete from NBrightBuy where typecode = 'CATEGORYLANG' and ParentItemId not in (Select itemid from NBrightBuy where typecode = 'CATEGORY')