

Open Catalog Interface (OCI):

Open Icecat XML and Full Icecat XML Repository

Revision date: February 27, 2014, Version 2.44

IMPORTANT NOTES

1. In case of Open Icecat, please check the Open Icecat sections.
 2. The links and files in the download directories are to be downloaded with scripts (from server to server). If you try to download them via your browser, you may sometimes experience memory problems. This does not mean that on our side the data is not accessible or correctly available. In case you experience any problem, contact us via the contact form.
 3. By default a daily index file is shown when you only enter the directory name. This, because it is more efficient for testing and viewing in a browser. To get the complete index file with active* products for Open Icecat users and all described products for full Icecat users, include in the path the full file name of the complete index: files.index.xml (in Open Icecat or Full Icecat).
- * products that are on_market minus 3 months or products that are created in the past 12 months.
4. After a rip of our general site, an IP address may be temporarily blocked. A rip is not acceptable as it may hurt site performance and can be a copyright infringement.

Table of contents

1. Subject of this document	4
Our mission	4
International Standard Supported	4
Open (Icecat) Repositories	4
Open Icecat Fair Use Policy	4
Coverage analysis	4
Advantages of Upgrading to Full Icecat.....	5
2. Directories	6
2.1 Access	6
2.2 Individual Product XML (Meta) Requests (Real-Time)	6
2.3 Open Icecat (free) directories, for batch processing	7
2.4 Full Icecat directories, for batch processing.....	7
2.5 Languages Supported and their Codes.....	8
2.6 Use Gzip / mod_deflate.....	9
2.7 Use of HTTP like FTP, and an example C# script to download files	9
3. Index files *.index.xml.....	11
3.1 Purpose	11
3.2 *.index.xml DTD	12
4. Additional data	14
4.1 Manufacturer names mapping data.....	14
4.2 Manufacturer part number mappings	14
4.3 References	14
4.3.1 Measures list (units)	16
4.3.2 Features list	16
4.3.3 Categories list	16
4.3.4 Suppliers list request (manufacturers)	17
4.3.5 Category features list request	17
4.3.6 Supplier product families list request.....	19
4.3.7 Languages list	19

4.3.8 Relations list	19
4.3.9 Campaigns list.....	20
4.3.10 Popularity of products.....	20
4.3.11 Standardized Product Summary Description	20
5. Product XML data file	22
5.1 Repository file DTD.....	22
5.2 XML Schema definitions	22
5.3 Example product XML	22
5.4 Product XML useful diagrams.....	23
5.5 How to use videos that are hosted by Icecat?	25
5.6 Feature logos via Icecat XML.....	25
6. SQL set-up.....	27
6.1 Explanation of entities.....	27
6.2 MySQL Set-Up (monolingual catalogue)	27

1. Subject of this document

This document describes the Open Catalog Interface (OCI): a set of standards and file formats used in the Icecat XML repository. It gives the basics of how to use the Icecat repository or how to publish data to Icecat conforming the OCI.

Our mission

It is our mission to provide channel partners and buyer orientation websites with high-quality product information via our Open Catalog Interface (OCI). More and more manufacturers work together with Icecat to take care of their product content; this makes it possible for us to distribute their product content for free. Here, you can always find the latest overview of sponsoring (free) brands in Open Icecat: <http://www.icecat.biz/en/menu/partners/index.htm>.

However attractive the free Open Icecat database is, the Full-Icecat database still contains many, many more brands: 5000+ (see <http://www.icecat.biz/en/menu/vendors/index.htm>). If you are interested in subscribing to the Full-Icecat database, please contact us via the website. If you want specific brands to be included into free Open Icecat: please, recommend Icecat to your manufacturer contacts.

International Standard Supported

XML for the exchange of catalog data
DTD and XSD for the definition of XML schema
The encoding used in our interfaces is UTF8
HTTP for file transfer
Gzip, Bzip2 and Zip for file compression
UNSPSC (see <http://www.unspsc.org>) for categorization
GTIN / EAN / UPC for logistical product codes (when provided by manufacturers)
ISO 639-1 two character code for languages
ISO 3166-1 two character code for countries
All common graphical formats for images (JPEG, GIF, ...)
Diverse product-specific standards, as mentioned in the feature descriptions.

Open (Icecat) Repositories

The *Full Icecat XML repository* is a set of product data files and an index file, which includes all pointers to individual product files and gives some basic information like time stamp updated, content quality level, etc. The Icecat XML repository contains product information for products of *all* brands, and categories that subscribers want us to support. The current content can be best reviewed on the public website: <http://www.icecat.biz/>.

The *Open Icecat repository* is the part of the Icecat XML repository that is sponsored by manufacturers and is provided for free to the channel. Open Icecat is distributed under the Open Content License Agreement: <http://icecat.biz/forum.cgi?post=3004>. Which brands are included in Open Icecat? The most recent list can always be found on the Icecat.biz site: <http://www.icecat.biz/en/menu/partners/index.htm>.

Open Icecat is also useful during development and for testing purposes.

Open Icecat Fair Use Policy

The user of Open Icecat product data (in a free Open Icecat subscription or as part of the Full Icecat data) is required to respect the Open Icecat Fair Use Policy, which can be found in the forum:
http://Icecat.biz/forum.cgi?tmpl=view_message.html;message_id=3004;thread_id=17

Explicitly relevant for website development is:

- to mention explicitly the "(c) Icecat.biz" copyright notice (or "Source: Icecat.biz") on all product data sheets, with a link to the Icecat.biz web site;
- include explicitly the Icecat "AS IS" disclaimer which can be found here:
<http://www.icecat.biz/en/menu/disclaimer/index.htm> (and is available in several languages).

Coverage analysis

To decide to make use of Full Icecat or Open Icecat it may be important to analyze the coverage of your portfolio or the portfolio of your supplier. You can always ask us to give you an overview of the coverage of (stock) portfolio of your suppliers (i.e., distributors) at any given moment. We strive to cover at least 90% of the stock portfolio (stock > 1) of mainstream distributors and work together with partners to monitor this coverage daily.

If you want to analyze coverage, you can do two things:

1) Register and upload your catalog to Icecat via the Login. In the match analysis you get the match percentages back for Full Icecat and Open Icecat. And you can get a list of the unrecognized products. An advantage is that you can also optionally display your offers -using this method- on the Icecat.biz price comparison sites.

NOTE: if you follow this method, Icecat also provides you automatically with your private Icecat index

2) If you do in-depth coverage analyses, please use the following matching file, which is a plain dump of everything in our catalog: https://data.icecat.biz/prodid/prodid_d.txt.gz (a very large file with all known products in the Icecat database, described and standardized or not in gzipped archive).

NOTE: Market presence=Y in the matching file indicates that we notice that the product is currently present in at least one distributor's catalog. Quality=Icecat means that Icecat editors have standardized the data-sheet. Quality=SUPPLIER means that the data-sheet is auto-imported from a manufacturer's CMS, and may be queued for standardization. Quality=NOEDITOR means that our editors did not create a data-sheet yet, nor did we auto-import manufacturer data; so, it is only the raw data as we received it from channel partners.

TIP: discuss with the Icecat team a service level for your portfolio.

Advantages of Upgrading to Full Icecat

What are the main advantages of Full Icecat over Open Icecat?

All product data of all 5000+ brands (see <http://www.icecat.biz/en/menu/vendors/index.htm>; if you see the brand logo the brand's product data is also available in free Open Icecat).

Coverage guarantees when we monitor your stock portfolio

Support for producing missing data-sheets, categories and brands on request

If you want to upgrade to Full Icecat, you can contact us via the contact form, or click on the "request Full Icecat access" link after the User Login (<http://www.icecat.biz/en/menu/openicecat/index.htm>), or fill in and fax the Full Icecat registration form:

http://icecat.biz/forum.cgi?tmpl=view_message.html;message_id=3191;thread_id=1

that you can find as an attachment. You will also find in the attached form (pdf) the standard Full Icecat tariffs.

2. Directories

2.1 Access

To obtain access to the repository, you will need a login/password, provided to you by Icecat. If you don't have these, please register online (click on Login) for free Open Icecat.

NOTE: During the registration you need to choose for the URL or XML version. If you want to make use of XML, make sure that you choose the Open Icecat XML subscription. You can always change this afterwards, or create a second subscription for the URL (links to hosted data-sheets) method.

For Full-Icecat access a separate contract is needed. You can request this contract form via info@icecat.biz.

Take care that you have provided us with the right IP addresses of your content servers, as the access is secured. You can test the login/password combination by entering it by hand in the BASIC HTTP authorization pop-up screen, which appears automatically when the directory URL is entered. In this case, we need to have added the IP address of your workplace as well to your profile. If you don't know this IP address, just type in your browsers <http://www.whatismyip.com/> or <http://www.getip.com/> and fill it in the IP addresses field after Login via the Icecat website (multiple IPs need to be entered space separated).

Of course, for professional use, server access need to be set up. In case of little experience in this field, we advise you to make use of the URL versions (links to pictures and datasheets) or ask us for assistance.

TIP: for osCommerce – a popular *open source* webshop based on PHP and MySQL – a free Icecat interface is available, which can be downloaded and adapted for your special needs. Here you can find the osCommerce connector, that is supported by <http://www.bintime.com/>:
<http://www.oscommerce.com/community/contributions,5294>

It is said that this interface works for ZenCart – a sibling of osCommerce – as well. Look in our Forum for interfaces to other *open source* solutions like Magento and Batavi, or proprietary ecommerce solutions.

2.2 Individual Product XML (Meta) Requests (Real-Time)

For smaller catalogs and for testing XML with small data-sets, it may be handy to make use of our real-time interface (URL2XML) for retrieving Product XML from the (Open) Icecat repositories.

The general format of the call is:

`http://data.icecat.biz/xml_s3/xml_server3.cgi?prod_id=<prod_id>;vendor=<vendor_name>;lang=<lang>;output=<output>`

or

`http://data.icecat.biz/xml_s3/xml_server3.cgi?ean_upc=<EAN or UPC>;lang=<lang>;output=productxml`

Where:

`prod_id` – the manufacturer part number of the requested product,
`vendor_name` – name of the manufacturer of the requested product,

ean_upc – the EAN or UPC code to identify a product,
lang – the language code (INT – for the international/standardized repository, EN, NL, FR... - for local ones, see section 2.4),
output – the type of the response. It can be productxml (the product xml file) and metaxml (the index or meta information on the Brand + ProdID/M_Prod_ID or EAN/UPC).

For example, if you need to get an xml file of product 'Compaq 6710b Base Model Notebook PC'.

Manufactured by HP, with part number RJ459AV, in English, the call will be:

https://data.icecat.biz/xml_s3/xml_server3.cgi?prod_id=RJ459AV;vendor=hp;lang=en;output=productxml

In case you want to retrieve data on the basis of an EAN or UPC code.

https://data.icecat.biz/xml_s3/xml_server3.cgi?ean_upc=4960999358246;lang=en;output=productxml

An example of the English standardized version of Canon 1447B006:

https://data.icecat.biz/xml_s3/xml_server3.cgi?prod_id=1447B006;vendor=Canon;lang=en;output=productxml

The meta data:

https://data.icecat.biz/xml_s3/xml_server3.cgi?prod_id=1447B006;vendor=Canon;lang=int;output=metaxml

The Dutch version (with for some brands localized data-sheets):

https://data.icecat.biz/xml_s3/xml_server3.cgi?prod_id=1447B006;vendor=Canon;lang=nl;output=productxml

The German version (with for some brands localized data-sheets):

https://data.icecat.biz/xml_s3/xml_server3.cgi?prod_id=1447B006;vendor=Canon;lang=de;output=productxml

XML Server 3 access works via the basic HTTP authorization, also used for other HTTP requests.

TIP: Use the Presentation_Value parameter in localized data-sheets for display of a data-sheet, and use the standardized Value parameter for search & compare on standardized values.

TIP 2: LocalValue is a transformation of the international Value according to the local dictionary and the applicable Metric/Imperial value system. Especially, for local applications, we advise to use LocalValue for search & compare functionalities.

2.3 Open Icecat (free) directories, for batch processing

Open Icecat users have access to:

<https://data.icecat.biz/export/freexml/EN/> for access to the standardized data files (QUALITY=Icecat).

TIP: The standardized files are advised for searching & comparing.

Please, note, where QUALITY = SUPPLIER only the original product data as imported from a manufacturer's CMS is available, the respective data-sheet is not yet standardized by our editors. When it is standardized the parameter is changed to QUALITY = Icecat.

The language-specific data-files are found here:

http://data.icecat.biz/export/freexml/<code>/<product_id>.xml, where <code> stands e.g. for NL, EN, FR, DE, IT, ES, DK etc. For the used codes see section 2.4.

Please, note that this language-specific content contains the standardized content (sames as in INT), which is entered by our editors, or – if no standardized data is yet available – just the original data imported from the CMS of a manufacturer like HP, Toshiba, Philips etc.

In general, we follow the ISO 639-1 two character country codes, except when a code can not be unambiguously tied to a language. So we use for example EN_SG, and EN_IN to identify Singapore English and Indian English respectively.

TIP: We advise the language-specific product content (Presentation_Value) purely for the display of product data-sheets, not for search & compare. For search & compare, we advise the standardized product content of the Value parameter from the /INT or local language directory. We advise LocalValue for country-specific search & compare applications.

TIP: Open Icecat is also useful for testing purposes before you upgrade to Full Icecat. The technical structures are 100% compatible with those of Full Icecat.

2.4 Full Icecat directories, for batch processing

Full Icecat contains all product data of all 5500+ brands supported. Often provided with service levels and coverage guarantees.

For the Full Icecat subscribers, a separate directory structure is in place. For example:

<http://data.icecat.biz/export/level4/EN>

Please, note, where QUALITY = SUPPLIER only the original product data as imported from a manufacturer's CMS is available, the respective data-sheet is not yet standardized by our editors. When it is standardized the parameter is changed to QUALITY = Icecat.

and other language versions are found here:

http://data.icecat.biz/export/level4/<code>/<product_id>.xml, where <code> stands e.g. for NL, FR, DE, IT, ES, DK, etc. For the full list of used codes see section 2.4.

Please, note that this language-specific content is the standardized content, which is entered by our editors, or – in case of no standardized values – the original data imported from the CMS of a manufacturer like HP, Toshiba, Philips etc.

TIP: We advise the language-specific product content (Presentation_Value parameter) purely for the display of product data-sheets, not for search & compare. For that, we advise the standardized product content (the Value parameter) in both the /INT and the respective local language directories. For country-specific applications we advise LocalValue in stead of the international Value.

Examples:

<https://data.icecat.biz/export/level4/EN>

<https://data.icecat.biz/export/level4/NL>

<https://data.icecat.biz/export/level4/FR>

<https://data.icecat.biz/export/level4/DE>
<https://data.icecat.biz/export/level4/IT>
<https://data.icecat.biz/export/level4/ES>
<https://data.icecat.biz/export/level4/DK>
<https://data.icecat.biz/export/level4/SE>
<https://data.icecat.biz/export/level4/RU>
<https://data.icecat.biz/export/level4/CZ>
<https://data.icecat.biz/export/level4/ZH>
<https://data.icecat.biz/export/level4/FI>
<https://data.icecat.biz/export/level4/PL>
<https://data.icecat.biz/export/level4/BG>
<https://data.icecat.biz/export/level4/TR>
etc

In general, we intend to follow the ISO 639-1 two character code.

2.5 Languages Supported and their Codes

If you want to know if your country/language is supported look here online for the latest status:

<http://www.icecat.biz/index.cgi?language=en;menu=country|language>

We currently support most world languages and European languages. Our intention is to support eventually all languages.

In general we follow the language codes as given by the ISO 639-1 standard. See

http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes. However, in some cases, we will choose for the two character country codes when for the “country-specific dialect” there is no language code given (e.g., BR for Brazilian-Portuguese, or US for US English).

In total we support now more than 20 world languages. If you feel that your language or the language that you need is missing, feel free to contact us. Probably we can work something out.

Used codes in Icecat:

INT – International standardized version of a data-sheet. When QUALITY = Icecat language independent values.
EN – Standard or UK English
US – US English
NL – Dutch
FR – French
DE – German
IT – Italian
ES – Spanish
DK – Danish
RU – Russian
PT – Portuguese
ZH – Chinese (simplified)
SE – Swedish
PL – Polish
CZ – Czech
HU – Hungarian

FI – Finnish
NO – Norwegian
TR – Turkish
BG – Bulgarian
KA – Georgian
RO – Romanian
SR – Serbian
JA – Japanese
UK – Ukrainian
CA – Catalan
HR – Croatian
AR – Arabic
VI – Vietnamese
HE - Hebrew
ZH_TW – Chinese (traditional)
BR – Brasilian Portuguese
KO – Korean
EN_SG – Singapore English
EN_IN – Indian English
LT – Lithuanian
LV – Latvian
DE_CH – Swiss German
ID – Indonesian
SK – Slovakian

If you need support for an additional language, don't hesitate to contact us

2.6 Use Gzip / mod_deflate

As files becomes bigger and bigger, and the number of files is growing fast, we support the gzipping of all interface files. Instead of doing this per file individually, we applied gzipping to all interface files automatically (mod_deflate). To benefit from it, you have to add one parameter to your calls.

About mod_deflate:

mod_deflate transports our interface files as compressed gzip data, but will show it transparently in its original format in modern browsers.

To benefit in your server scripts from gzip, you must use in any HTTP request this additional parameter:

Accept-Encoding: gzip

It will enforce gzipping!

Example:

GET /export/level4/EN/10.xml HTTP/1.1

Host: prf.icecat.biz

Accept-Encoding: gzip

User-Agent: Firefox/1.0

2.7 Use of HTTP like FTP, and an example C# script to download files

We often get questions like can we use FTP instead. Actually, this is really not necessary as http has very similar capabilities to FTP. One can use the following format: `http://username:password@sitename`

Here is example code in C# to download files (it can be used also in ASP.NET code). It is tested and works ok:

```
public static Byte[] DownloadIcecatFile()
{
    string strDownloadURL = "http://data.Icecat.biz/export/freexml/EN/480237.xml";
    string strUser = "Your Login to Icecat Repository";
    string strPWD = "Your Password to Icecat Repository";
    // Creating an instance of a WebClient
    WebClient req = new WebClient();
    // Creating an instance of a credential cache,
    // and passing the username and password to it
    CredentialCache cache = new CredentialCache();
    cache.Add(new Uri(strDownloadURL), "Basic", new NetworkCredential(strUser, strPWD));
    req.Credentials = cache;
    Byte[] fileData = req.DownloadData(strDownloadURL);
    return fileData;
}
```

In Classic ASP, MSXML2.ServerXMLHttp can be used
Some details are present here: <http://msdn.microsoft.com/en-us/library/ms766431%28v=vs.85%29.aspx>

3. Index files *.index.xml

3.1 Purpose

The purpose of the index files is to find the right Icecat number and thus product xml files for a given product. The “key” is always Brand + Manufacturer Part Number.

New:

We added also EAN / UPC codes as unique identifiers.

Also the list of distributor part numbers is present if available.

Plus an indicator if a product is on the market still or obsolete.

And the markets/countries where we daily see that the product is live.

There are four types of index files in our XML-OCI:

- an index file with references to active* product data-sheets in Open Icecat and with references to all described product data-sheets in Full Icecat (/files.index.xml or .csv)

* products that are on_market minus 3 months or products that are created in the past 12 months.

- a smaller index file with only references to the new or changed product data-sheets of the respective day (*/daily.index.xml or .csv)
- an index file with only the products that are currently on the market, as far as we can see that based on 100s of distributor and reseller price files (*/on_market.index.xml or .csv)
- an index file with the products that are or were on the market for which we only have basic market data, but no complete data-sheet (*/nobody.index.xml or .csv)

The complete Full Icecat index file with all described products for English is located at

<https://data.icecat.biz/export/level4/EN/files.index.xml.gz> (or .csv)

Historic/obsolete products can be downloaded via the real-time interface

or per language code: <https://data.icecat.biz/export/level4/<code>/files.index.xml.gz> (or .csv) ,

Example: <https://data.icecat.biz/export/level4/NL/files.index.xml.gz>, where <code> stands e.g. for FR, DE, IT, ES, DK, RU, ES, SE,... For the complete list see section 2.4.

Similarly the index file with changed or new product references for English is found at

<https://data.icecat.biz/export/level4/EN/daily.index.xml.gz> (or .csv.gz)

or per language code: <http://data.icecat.biz/export/level4/<code>/daily.index.xml> , where <code> stands e.g. for NL, FR, DE, IT, ES, DK, RU, SE,... For the complete list see section 2.4.

or in case of Open Icecat the index file with active products can be found here:

<https://data.icecat.biz/export/freexml/files.index.xml.gz> or

<http://data.icecat.biz/export/freexml/<code>/files.index.xml>, where <code> stands e.g. for NL, EN, FR, DE, IT, SP, DK, RU, ES, SE,... For the complete list see section 2.4.

* products that are on_market minus 3 months or products that are created in the past 12 months.

Similarly the index file with changed or new product references is found at

<https://data.icecat.biz/export/freexml/daily.index.xml.gz> or per language code:

<https://data.icecat.biz/export/freexml/<code>/daily.index.xml.gz> , where [code] stands e.g. for NL, EN, FR, DE, IT, SP, DK, RU, ES, SE, ... For the complete list see section 2.4.

It includes the index information for each product and gives the path to the xml product file. Below is an example of a file tag in the index file:

```
<file path="export/level4/INT/344.xml" Product_ID="344" Updated="20091031192020" Quality="Icecat" Supplier_id="29"
Prod_ID="WD400BB" Catid="219" On_Market="1" Model_Name="WD Caviar 40GB EIDE, 100 MB/s, 2 MB, 7200 RPM"
Product_View="25508">
<M_Prod_ID>WD400BB?1PK</M_Prod_ID>
<M_Prod_ID>WD400BBRTL2</M_Prod_ID>
<EAN_UPCS>
<EAN_UPC Value="0718037719696"/>
<EAN_UPC Value="2000006035103"/>
<EAN_UPC Value="4005922164744"/>
<EAN_UPC Value="5400853050002"/>
<EAN_UPC Value="8032976016497"/>
</EAN_UPCS>
<Country_Markets>
<Country_Market Value="AT"/>
<Country_Market Value="DE"/>
<Country_Market Value="ES"/>
<Country_Market Value="FR"/>
<Country_Market Value="IT"/>
<Country_Market Value="NL"/>
<Country_Market Value="SE"/>
<Country_Market Value="UK"/>
<Country_Market Value="US"/>
</Country_Markets>
</file>
```

The On_Market attribute indicates that a product is still on the market (On_Market="1") or not seen on the market anymore (On_Market="0"). Per product also the EAN or UPC codes that are used for logistical purposes are given. It is possible that there are multiple EANs or UPCs, because depending on packaging or importers different codes are given. Further, the part number variants that are used in the market are given in a sub list (M_Prod_ID).

Finally, the <Country_Market Value="US"/> indicates that we see the product at the moment in the catalogs of US channel partners. This tag can be useful to limit your view on the index file, in case that you don't have distributor imports to do that.

The new Model_Name and Product_View attributes were added. The Model_Name attribute contains the product name, the Product_View attribute indicates how many times the current product was requested.

files.index.xml and daily.index.xml are refreshed every day by us. You can use daily.index.xml to update only the data in your local database that has changed or is new in the Icecat database. files.index.xml, you can use the first time for a full import of needed product information of active data-sheets for Open Icecat users and all existing data-sheets for Full Icecat users, and for the regular analysis of coverage of your product database. Historic/obsolete products can be downloaded via the real-time interface.

* products that are on_market minus 3 months or products that are created in the past 12 months.

TIP 1: download ONLY the xml files that you actually need for your shop or application by using the daily export from your ERP or shop (or actually the daily imports from your distributor/suppliers) as a filter. It is useless to import all the EMEA/Worldwide data files, if you only operate in one geographic area or have an assortment limited to a certain category.

TIP 2: check the timestamps to only download files that have changed or check only for new files, to improve update performance cycles.

TIP 3: use the on_market index file, in case that you think the full index file is too big for you to process or too inefficient. DISCLAIMER: if our monitor for your market is incomplete, the on_market index file is probably too incomplete for your purposes.

TIP 4: in case that you want to recategorize your products, for which we have no data-sheet (QUALITY=NOEDITOR), make use of the nobody index file.

TIP 5: the daily index file also includes the list of product codes that are removed the day before (QUALITY=REMOVED). Products can be removed, for example, when product are mapped away. You can use it for cleansing your catalog.

3.2 *.index.xml DTD

The latest version of the DTD can be found here: <http://data.icecat.biz/dtd/files.index.dtd>

```
<!ELEMENT Icecat-interface (files.index)+>
<!ATTLIST Icecat-interface
      xmlns:xsi   CDATA #IMPLIED
      xsi:noNamespaceSchemaLocation CDATA #IMPLIED>

<!ELEMENT files.index (file)*>
<!ATTLIST files.index
      Generated CDATA #REQUIRED>

<!ELEMENT file (M_Prod_ID|EAN_UPCS|Country_Markets|Distributors)*>
<!ATTLIST file
      path          CDATA #REQUIRED
      Product_ID    CDATA #REQUIRED
      Updated       CDATA #REQUIRED
      Quality       CDATA #REQUIRED
      Supplier_id   CDATA #REQUIRED
      Prod_ID       CDATA #REQUIRED
      Catid         CDATA #REQUIRED
      On_Market     CDATA #IMPLIED
      Model_Name    CDATA #IMPLIED
      Product_View  CDATA #IMPLIED>

<!ELEMENT M_Prod_ID (#PCDATA)>

<!ELEMENT EAN_UPCS (EAN_UPC)+>

<!ELEMENT EAN_UPC EMPTY>
<!ATTLIST EAN_UPC
      Value        CDATA #REQUIRED>

<!ELEMENT Country_Markets (Country_Market)+>

<!ELEMENT Country_Market EMPTY>
<!ATTLIST Country_Market
      Value        CDATA #REQUIRED>
```

Each product has its own data file in xml format. Its location is defined by the attribute "path" in the *.index.xml.gz. The file content includes all the available product information for the respective product.

The "Prod_ID" attribute is a manufacturer's unique identifier for a product, often called Manufacturer Part Number (code de fabricant, Produkt-Kode, artikelnummer, etc). In combination with the "supplier_id" it is the unique key to select a product's XML file via Icecat's internal identifier ("Product_id").

M_Prod_ID are one or more manufacturer part numbers which were used by distributors/channel partners, but are mapped away to the original (correct) Prod_ID.

The EAN_UPC sub-list lists the GTIN codes (EAN or UPC) that are connected to this product. There can be multiple codes, as these are logistical codes depending on packaging types and geography.

On_Market (1=Yes, 0=No) indicates whether a product is somewhere seen on the market by Icecat. The Country_Market lists the individual markets (countries), where Icecat sees that the product is present. This indication can be used to limit the view on our index file.

DISCLAIMER: Our market view is always imperfect as we depend on the data of connected channel partners and distributors.

Catid is Icecat's internal category identifier. Catid="151" refers to Notebooks. For every category there is a second, external, category identifier based on UNSPSC. In the reference files or the product xml the meaning of categories can be found.

"Quality" attribute values:

Value	Explanation
SUPPLIER	The content is received from a supplier CMS, but not standardized by an Icecat editor. The language-specific directories are likely to contain the full (unstandardized) data-sheet.
Icecat	The content is entered or standardized by Icecat editors. The standardized data can be found in the INT directory <i>and</i> the language-specific directories.
NOEDITOR	The content is received from a merchant (in most cases one of the 100s of distributors we are daily "polling") and may be parsed. Editors haven't described this product yet. The NOEDITOR data is <i>not</i> exported in XML to 3 rd parties.

4. Additional data

4.1 Manufacturer names mapping data

To help you with matching distributor data to the Icecat rich content, we are publishing the mappings of used variants of manufacturer names to the Icecat Supplier_ID.

You can find this file in your repository, named `supplier_mapping.xml`.

Within this file you'll find the match between different manufacturer names variations and the original manufacturer name.

The format is self explanatory. The DTD for the message format is available at:

http://data.icecat.biz/dtd/ICECAT-supplier_mapping.dtd

4.2 Manufacturer part number mappings

Different distributors often produce different manufacturer part numbers (MPN) for the same product.

To help you correct the MPNs from different distributors and to help you to get the best coverage, we are publishing the available MPN mappings to the official MPN.

In every repository directory, the file with product mappings is published in the index files (files.index.xml etc.).

OBSOLETE METHOD `product_mapping.xml`:

INT/`product_mapping.xml`
EN/`product_mapping.xml`
etc...

We advise *not* to use `product_mapping.xml` files anymore as it has an obsolete structure, and all product mapping data is already integrated in the XML index files.

4.3 References

In product XML feeds, you will find a lot of structures, which are included as references in the product data files, e.g. categories, features, measures (units) etc.

The reference files can be found in the `/refs/` subdirectory. For example, for Full Icecat, at

<https://data.icecat.biz/export/level4/refs/> and for Open Icecat at

<https://data.icecat.biz/export/freexml/refs/>

The included files (entities) are:

CampaignsList.xml (with the documentation)

CategoriesList.xml.gz

CategoryFeaturesList.xml.gz

FeaturesList.xml.gz

LanguageList.xml.gz

MeasuresList.xml.gz

RelationsList.xml

SupplierProductFamiliesListRequest.xml.gz

SuppliersList.xml.gz

FeatureValuesVocabularyList.xml.gz (translations of certain frequent feature values)

Index of /export/freexml/refs

Name	Last modified	Size	Description
 Parent Directory		-	
 CampaignsList.xml	2009-11-06 04:14	34K	
 CampaignsListDocumentation.txt	2009-09-28 13:53	1.2K	
 CategoriesList.xml.gz	2009-11-06 04:05	972K	
 CategoryFeaturesList.xml.gz	2009-11-06 04:05	18M	
 FeatureValuesVocabularyList.xml.gz	2009-11-06 04:05	108K	
 FeaturesList.xml.gz	2009-11-06 03:59	3.5M	
 LanguageList.xml.gz	2009-11-06 04:05	3.3K	
 MeasuresList.xml.gz	2009-11-06 04:05	82K	
 RelationsList.xml	2009-11-06 04:14	237K	
 SupplierProductFamiliesListRequest.xml.gz	2009-11-06 04:05	613K	
 SuppliersList.xml.gz	2009-11-06 03:59	44K	

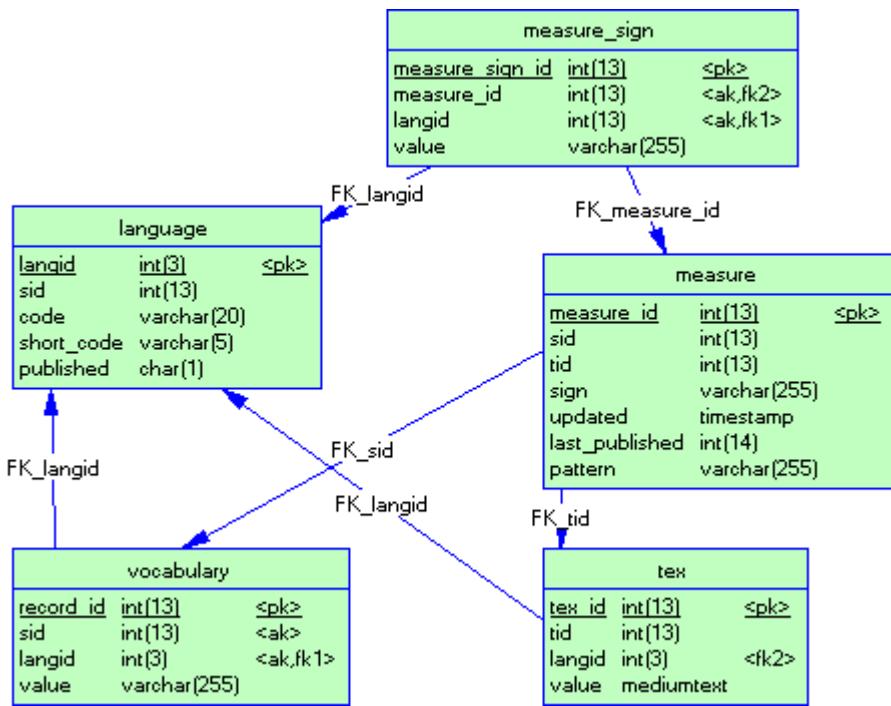
Almost all files are present in Gzipped format. Typical use of the files is in combination with data from the INT (international) directory, not the localized directories as there the language-specific values are already automatically applied to the product XML files.

Note: The single file with all references, refs.xml, is actually replaced by the above-mentioned directory, as the refs.xml file became too big to handle easily.

4.3.1 Measures list (units)

Example:

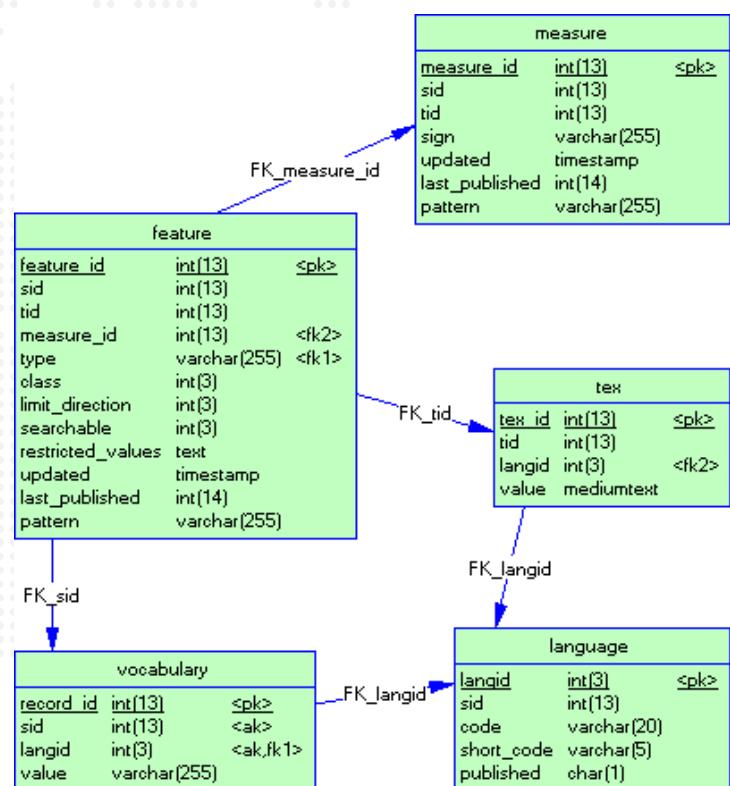
```
<Measure ID="57">
  <Signs>
    <Sign ID="40"
langid="1">GB/h</Sign>
  </Signs>
  <Names>
    <Name ID="2415"
langid="1">gigabyte/h</Name>
  </Names>
  <Descriptions>
    <Description ID="10564"
langid="1">desc</Description>
  </Descriptions>
  <Sign>GB/h</Sign>
</Measure>
```



4.3.2 Features list

Example:

```
<Feature ID="1630" Type="numerical" Class="0">
  <Measure ID="50" Sign="bit">
    <Signs>
      <Sign ID="34" langid="1">bit</Sign>
    </Signs>
  </Measure>
  <Names>
    <Name ID="5089" langid="1">Memory
bus</Name>
  </Names>
  <Descriptions>
    <Description ID="1557"
langid="1">desc</Description>
  </Descriptions>
</Feature>
```



4.3.3 Categories list

This chunk provides the information on categories which are used in the Icecat environment and for product categorization. Each category may be either searchable or not. Searchable categories may be used for product lookup by product feature values, as Icecat editors are systematically maintain data integrity to allow smart searches.

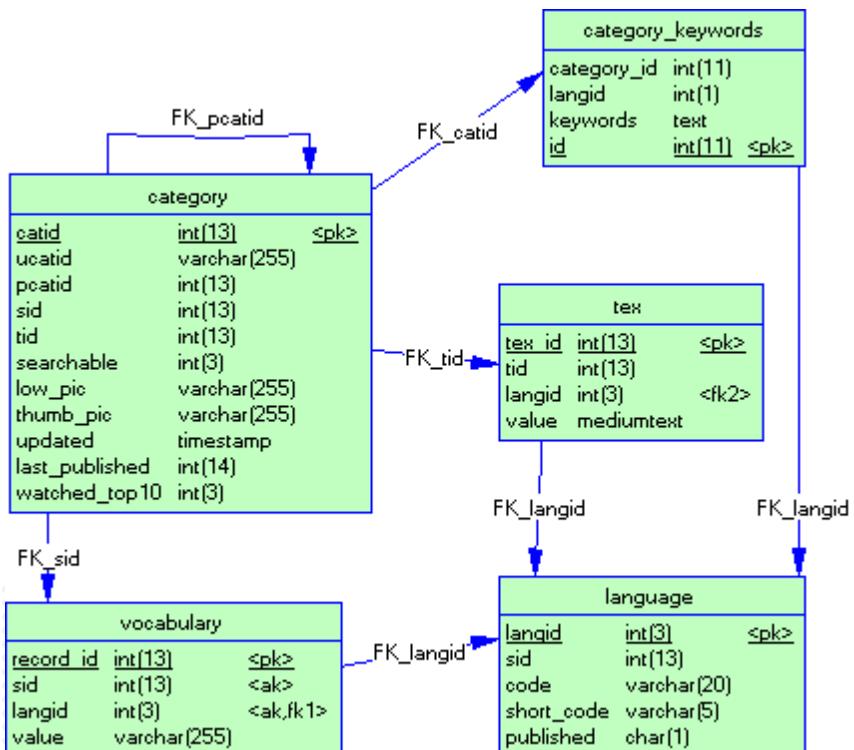
Explanation to values of the “Searchable” attribute:

Value	Explanation
1	This category may be used for product lookup in product list lookup request
0	This category is not made searchable (in our own product finder tools)

“Score” attribute in the response reflects the category usage statistic. The higher number means the higher usage level.

Example:

```
<Category ID="151"
UNCATID="43171801" Searchable="1"
```



```

ThumbPic="http://images.Icecat.biz/thumbs/CAT151.jpg" Score="17303447" LowPic="http://images.Icecat.biz/img/low_pic/151-5264.jpg">
<ParentCategory ID="150">
    <Names>
        <Name ID="301" langid="1">computers</Name>
    </Names>
</ParentCategory>
<Keywords ID="28" Value="laptop, laptops, notebook, notebooks, portable PC" langid="1"/>
<Name ID="303" Value="notebooks/laptops" langid="1"/>
<Description ID="233" Value="A notebook, also known as laptop..." langid="1"/>
</Category>

```

4.3.4 Suppliers list request (manufacturers)

Example:

```
<SuppliersList>
```

```
<Supplier Name="Hitachi" ID="169"/>
<Supplier Name="Lexmark" ID="20"/>
<Supplier Name="Xerox" ID="30"/>
/>/SuppliersList>
```

4.3.5 Category features list request

This chunk provides information about features which are assigned to a every certain category and which are used for describing products in the category.

Each searchable feature will have LimitDirection attribute defined, which will give an advice for finding a “better” feature value, either `0` or `1`. E.g. for feature “Hard disk capacity” it would be `0`.relation, and for the feature “Write seek” it would be `1` relation.

“LimitDirection” attribute values explanation:

Value	Explanation
0	The relation is undefined
1	The relation is <code>0</code> .
2	The relation is <code>1</code> .
3	The relation is <code>=</code>

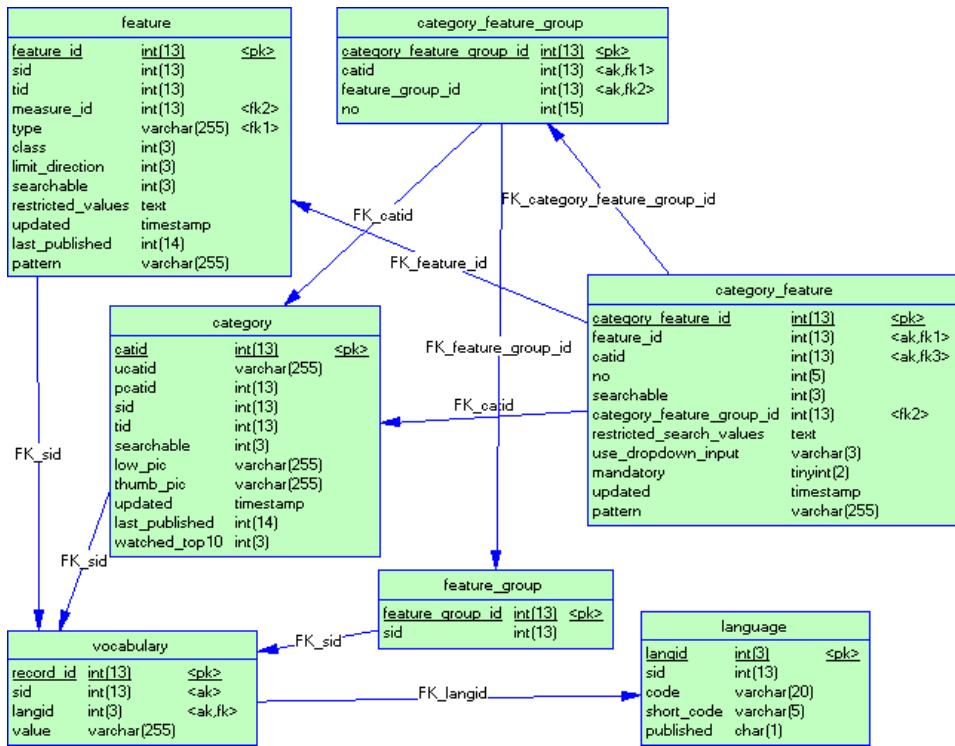
“Class” attribute values explanation:

Value	Explanation
0	Key feature
1	Extra feature – tech details

Possible feature values may be restricted to a limited set of value. In that case possible values for combination Category-Feature are listed into container RestrictedValue.

Example:

```
<Category ID="559" UNCATID="45111601" LowPic="">
  <CategoryFeatureGroup ID="607" No="0">
    <FeatureGroup ID="0">
      <Name ID="5073" Value="Technical details" langid="1"/>
    </FeatureGroup>
  </CategoryFeatureGroup>
  <CategoryFeatureGroup ID="4322" No="0">
    .....
  </CategoryFeatureGroup>
  <Feature ID="4169" CategoryFeature_ID="33330" CategoryFeatureGroup_ID="607" Searchable="0" LimitDirection="0"
  No="100000" Class="0">
    <RestrictedValue/>
    <Measure ID="54" Sign="m">
      <Signs>...</Signs>
    </Measure>
    <Name ID="59064" Value="Maximum range" langid="1"/>
  </Feature>
  <Feature ID="94" .....>
    .....
  </Feature>
  <Name ID="1119" Value="pointers" langid="1"/>
</Category>
```



4.3.6 Supplier product families list request

This chunk provides the list of supplier product families. Each product family may have information about it's Name, Description in different languages, and category.

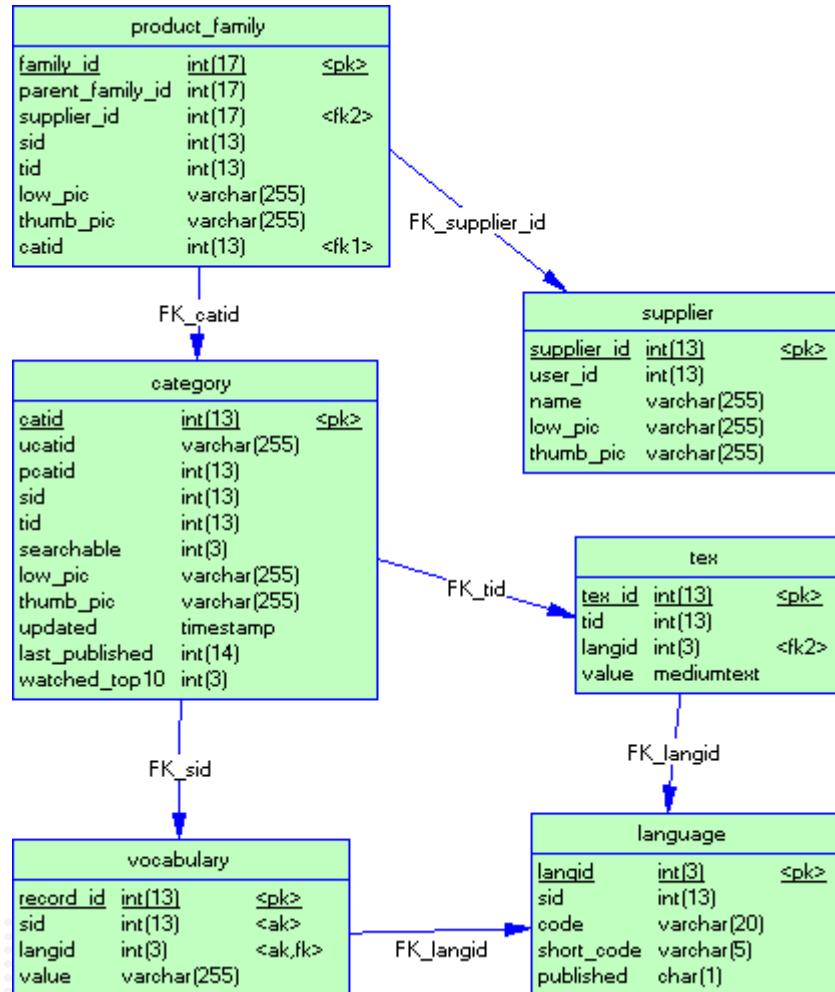
```
<ProductFamily Category_ID="151" ID="2"
ThumbPic="http://images.Icecat.biz/thumbs/FAM2.jpg"
LowPic="http://images.Icecat.biz/img/families/2-
3386.jpg">
<Supplier ID="7" Name="Acer"/>
<Description ID="6817" Value="Acer's whole range
of TravelMate mobile computers delivers superb
performance and reliable on-the-road convenience."
langid="1"/>
<Name ID="9512" Value="TravelMate series"
langid="1"/>
<ParentProductFamily ID="1"/>
</ProductFamily>
```

4.3.7 Languages list

Example:

```
<Language ID="6" Code="spanish" ShortCode="ES"
Sid="4797">
<Name ID="14094" Value="Spanish" langid="1"/>
<Name ID="14095" Value="Spanish" langid="2"/>
<Name ID="14096" Value="Spanish" langid="3"/>
<Name ID="18228" Value="Spanish" langid="4"/>
<Name ID="22360" Value="Spanish" langid="5"/>
<Name ID="26492" Value="Spanish" langid="6"/>
</Language>
```

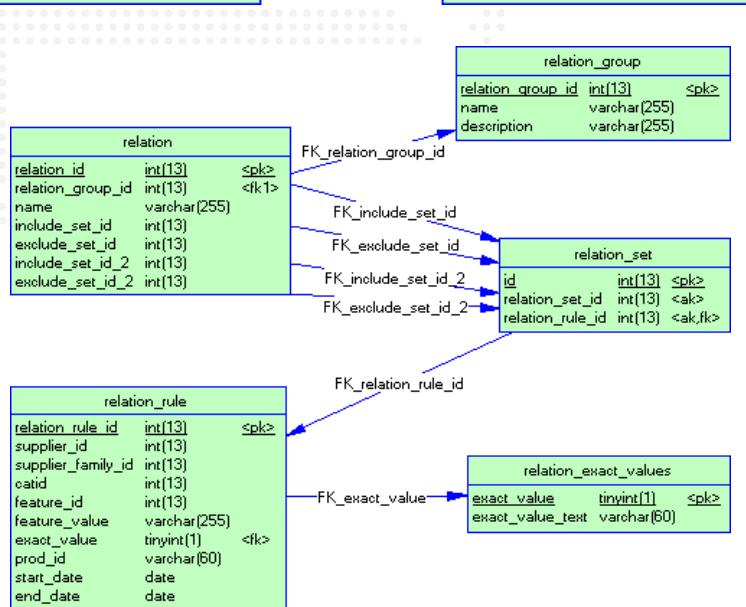
For the used language codes see section 2.4



4.3.8 Relations list

Icecat has the dynamic relations (x-sells). They based on the product sets, manually defined. One product set relates to another one.

```
<RelationGroup ID="5" name="Conceptronic"
description="Conceptronic">
<Relation ID="4" name="CLLSPK21">
<SourceIncludeRules>
<Rule ID="17">
<Prod_id value="CLLSPK21"/>
</Rule>
</SourceIncludeRules>
<DestinationIncludeRules>
<Rule ID="18">
<Category ID="153"/>
<Feature ID="2313" exact="more">0</Feature>
<Start_date value="2008-10-01"/>
<End_date value="2009-07-01"/>
</Rule>
</DestinationIncludeRules>
</Relation>
```

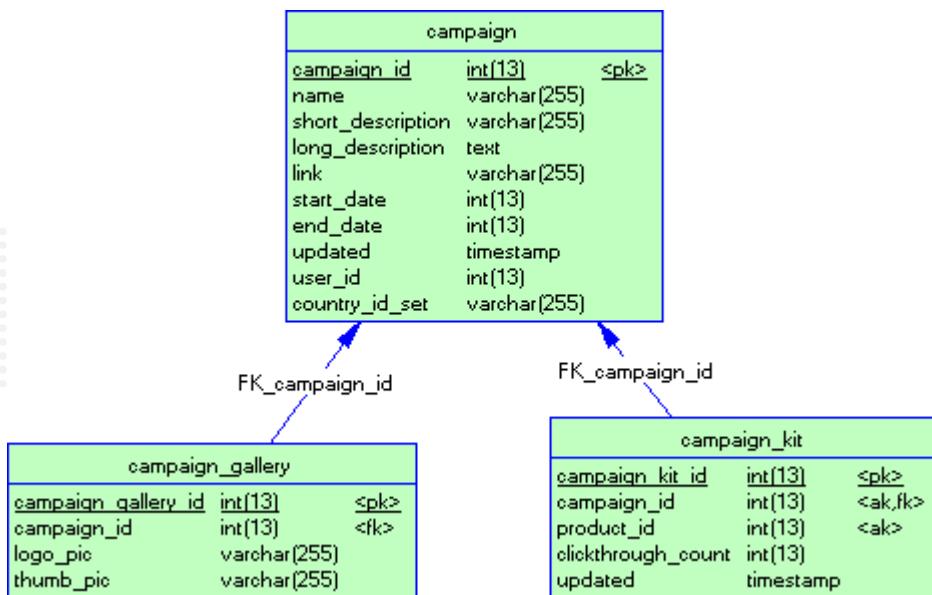


4.3.9 Campaigns list

Brand products can have campaigns for specific period. CampaignsList.xml has the list of all assigned campaigns.

```
<Campaign ID="8" User_ID="5" Name="25 jaar HP LaserJet" StartDate="1249077600" EndDate="1256943600" Updated="2009-10-12 11:17:36" Country="Netherlands" LandingPage="">
    <Title>25 jaar HP LaserJet = 25% korting!</Title>
    <Motivation>HP LaserJet bestaat 25 jaar,...</Motivation>
    <ImagesList>
        <Image ID="7" LogoPic="http://images.icecat.biz/img/campaign/7-3450.jpg"
        ThumbPic="http://images.icecat.biz/img/campaign_thumbs/7-3450.jpg"/>
    </ImagesList>
    <ProductsList>
        <Product ID="327685" ProId="Q7494A" Vendor="HP" Name="Color LaserJet 4700dtn Printer" Link="" Clicks="0"/>
    </ProductsList>
</Campaign>
```

4.3.10 Popularity of products



All Icecat product requests (real-time requests by end-users and xml requests) are logged. This data gives a product popularity indication which can be used to sort products and to analyze trends as input for marketing decisions. This popularity data is daily published on:

http://data.icecat.biz/export/level4/data_prod_stat.xml

4.3.11 Standardized Product Summary Description

In response to questions from different clients, we have created a Standardized Product Summary Description.

The English version is to be found here:

https://data.icecat.biz/export/level4/EN/product_overview.txt

https://data.icecat.biz/export/level4/EN/product_overview.txt.gz

Where column 1 is the Icecat ID, and column 2 and 3 give a short and a long standardized product summary description.

The short summary of column 2 is created as follows:

<brand name> + <Model name> + <product family, if present> + <key feature value 1..5 + unit>

The extended summary of column 3 is created as follows:

<brand name> + <Model name> + <product family, if present> +
<name feature group 1> + <related key feature values + units>
<name feature group ..> + <related key feature values + units>
<name feature group n> + <related key feature values + units>

The local versions can be found here:

http://data.icecat.biz/export/level4/<language_code>/product_overview.txt

http://data.icecat.biz/export/level4/<language_code>/product_overview.txt.gz

TIP: Think about the guarantees that you get from your suppliers (distributors), based on errors made in their product descriptions. It may still be smart to make use of these supplier (distributor) texts in the ordering process for this reason.

5. Product XML data file

5.1 Repository file DTD

The XML repository files comply to the standard Icecat XML response DTD, which is located at http://data.icecat.biz/dtd/ICECAT-interface_response.dtd

Please, note that the tag "No=" (example: `No="100090"`) is a Priority indicator. The higher the number the more important the feature or feature group is considered to be for buyer orientation. We advise you to sort features and feature groups by priority, to get a more customer friendly user presentation.

Product code is a deprecated field, earlier it used as product response status. Now, product code always equals 1.

5.2 XML Schema definitions

Each Icecat XML has its XSD definition, you can see it as 2 mandatory attributes in the root XML tag. The basic information about XML Schema - http://en.wikipedia.org/wiki/XML_Schema_%28W3C%29. Icecat XSD sources are located at <http://data.icecat.biz/xsd/>. They can be used for Icecat XMLs validation.

5.3 Example product XML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE Icecat-interface SYSTEM "http://data.icecat.biz/dtd/ICECAT-interface_response.dtd">
<!-- source: icecat.biz 2009 -->
<Icecat-interface xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://data.icecat.biz/xsd/ICECAT-interface_response.xsd">
    <Product ReleaseDate="2005-07-15" ID="269830" ThumbPicSize="3486" Quality="Icecat"
    HighPic="http://images.icecat.biz/img/norm/high/269830-1155.jpg" LowPic="http://images.icecat.biz/img/norm/low/269830-1155.jpg" LowPicSize="15138" Prod_id="LKN:NDL-152102-002" HighPicSize="115619"
    ThumbPic="http://images.icecat.biz/thumbs/269830.jpg" Code="1" Name="AMILO D-7850 P4-3.2G HT 538">
        <ProductDescription ID="304191" URL="http://www.fujitsu-siemens.com/products/mobile/notebooks/amillo_d.html"
        langid="1" ManualPDFURL="" ManualPDFSize="0" PDFURL="http://pdfs.icecat.biz/pdf/304191-8497.pdf" ShortDesc="AMILO D-7850 P4-3.2G HT 538 80GB 1024MB15 TFT MN70405 NL (P4)" LongDesc="Neither video freaks nor digital photographers need a desktop any longer. With this power-notebook in a backpack, advanced graphic features are available everywhere, on a flight, in a train or on the beach. The special ATI MOBILITY RADEON 9000 graphic card guarantees excellent 3D performance as well as smooth DVD-playback. The Intel Pentium 4 processor provides all the power you \nneed for your high-end applications. You can easily attach other devices through an astonishing range of multimedia interfaces including FireWire, S-Video and 5 USB 2.0 ports. The AMILO D, the truly portable powerhouse and mobile alternative to your desktop PC. " WarrantyInfo="" PDFSize="140328"/>
        <ProductMultimediaObject/>
        <ProductFeature Localized="0" ID="10476921" Value="3200" CategoryFeature_ID="50" CategoryFeatureGroup_ID="35"
        No="10105478" Presentation_Value="3200 MHz">
            <LocalValue Value="3200">
                <Measure ID="18">
                    <Signs>
                        <Sign ID="9" langid="1"><![CDATA[ MHz ]]></Sign>
                    </Signs>
                </Measure>
            </LocalValue>
            <Feature ID="5">
                <Measure ID="18" Sign="">
                    <Signs><Sign ID="9" langid="1"><![CDATA[ MHz ]]></Sign></Signs>
                </Measure>
                <Name ID="1291" Value="Processor clock speed" langid="1"/>
            </Feature>
        </ProductFeature>
    </Product>
</Icecat-interface>
```

```

</ProductFeature>
<ProductFamily ID="90">
    <Name ID="10161" Value="AMILO D" langid="1"/>
</ProductFamily>
<SummaryDescription>
    <ShortSummaryDescription>Fujitsu AMILO D-7850 P4-3.2G HT 538 AMILO D, 3200 MHz, Intel Pentium 4, Intel
Pentium IV, 533 MHz, SIS 648 + SIS 963L, 80 GB</ShortSummaryDescription>
    <LongSummaryDescription>Fujitsu AMILO D-7850 P4-3.2G HT 538, AMILO D. Processor: 3200 MHz, Intel Pentium 4,
Intel Pentium IV, 533 MHz, SIS 648 + SIS 963L. Disk drive: 80 GB, Ultra-ATA/133. Display: 15 &quot;, TFT XGA, 1024 x 768 pixels,
750:1. Memory: 1024 MB, PC/2700 DDR333 (Desktop memory). Video: ATI MOBILITY RADEON ? 9200, 64 MB. Optical drive:
DVD?RW Double Layer, 24 x, 24 x, 20, 10 x. Audio: VT1612, 2 built-in speaker, built-in microphone, microphone in, headphone out,
1 x volume regulator , 10/100 Mbps LAN, 56 Kbit/s. Operating system/software: Microsoft Works Suite 2004 including Word 2002,
Works 7.0,. Energy management: Lithium-Ion, 4000 mAh, 2 h. System requirements: Microsoft? Windows? XP.. Weight &amp;
dimensions: 3500 g, 345 x 282 x 41 mm</LongSummaryDescription>
</SummaryDescription>
<ProductBundled/>
<ProductRelated/>
<ProductGallery/>
<CategoryFeatureGroup ID="269" No="-1">
    <FeatureGroup ID="0">
        <Name ID="5073" Value="Technical details" langid="1"/>
    </FeatureGroup>
</CategoryFeatureGroup>
<EANCode/>
<Supplier ID="15" Name="Fujitsu"/>
<Category ID="151">
    <Name ID="303" Value="notebooks/laptops" langid="1"/>
</Category>
</Product>
</Icecat-interface>

```

Some notes:

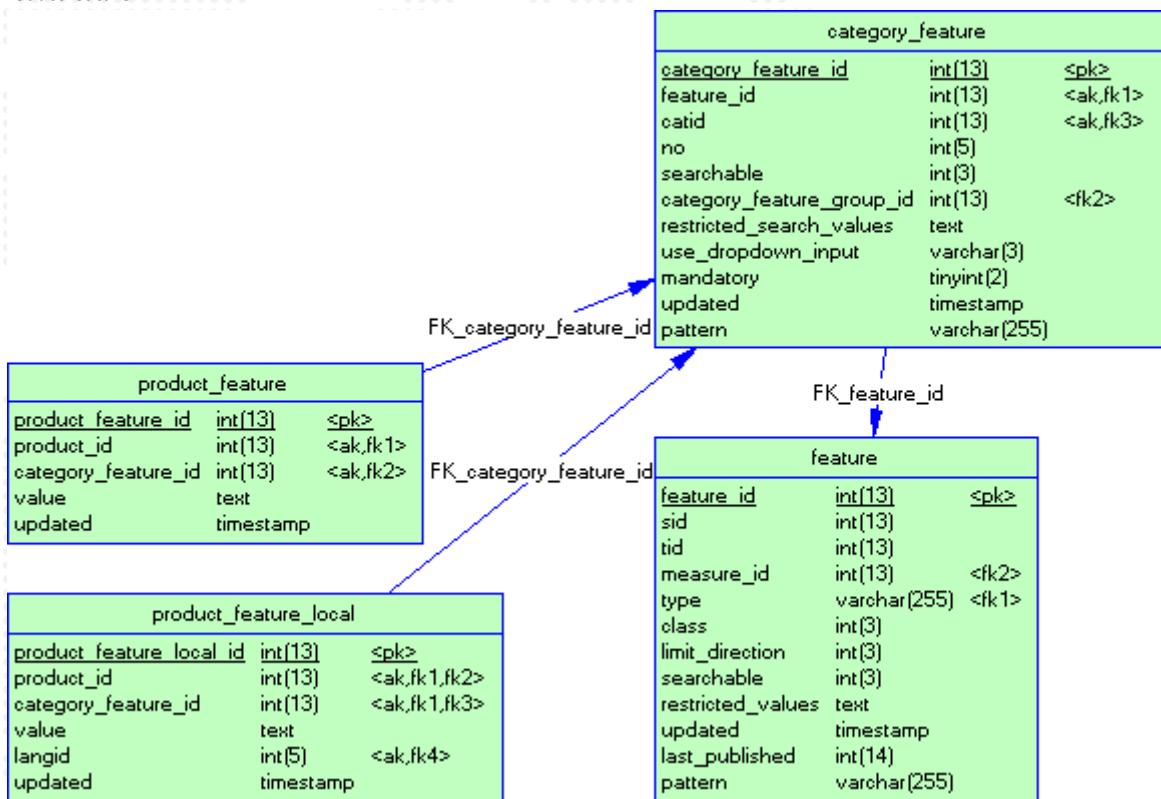
- the Reverse attribute for product relations is obsolete and can be ignored.
- Presentation_Value is the processed value as can be displayed in a data-sheet. It is based on the international (but localized) feature value with the localized measure unit and localized decimal separator (comma or dot depending on the country) or - if the international value is absent - on the language-specific value. Also it contains automatically transformed unit (for e.g. "665 mm is being transformed to more friendly "66,5 cm" or "66.5 cm" depending on the country).
- LocalValue is a local transformation of our international Value attribute. It applies the local dictionary to Value and transforms values according to local standards. As in Value attribute LocalValue contains value of the feature and measurement unit as a separate attributes.
- ProductSummaryDescription – the new data structure, collected from the product feature names & values. ShortSummaryDescription has: product name, product family and 1st 6 feature values. LongSummaryDescription has product name, product family and the list of feature group name with the feature values.

5.4 Product XML useful diagrams

```
<ProductDescription
ID="304191"
URL="http://..."
langid="1"
ManualPDFURL="http://..."
ManualPDFSize="0"
PDFURL="http://..."
ShortDesc="AMILO D-7850..."
LongDesc="Neither video freaks...
WarrantyInfo="..." PDFSize="140328"/>
```

product_description		
product_description_id	int(13)	<pk>
product_id	int(13)	<ak,fk1>
langid	int(13)	<ak,fk2>
short_desc	text	
long_desc	text	
official_url	varchar(255)	
warranty_info	text	
updated	timestamp	
pdf_url	varchar(255)	
pdf_size	int(13)	
manual_pdf_url	varchar(255)	
manual_pdf_size	int(13)	

```
<ProductFeature ID="10476921" Local_ID="0" Value="3200" CategoryFeature_ID="50" CategoryFeatureGroup_ID="35" No="10105478" Presentation_Value ="3200 MHz" Localized="0" Translated="0">
<LocalValue Value="3200">
<Measure ID="18">
<Signs><Sign ID="9" langid="1"><![CDATA[ MHz ]]></Sign>
</Signs>
</Measure>
</LocalValue>
<Feature ID="5">
<Measure ID="18" Sign="">
<Signs><Sign ID="9" langid="1"><![CDATA[ MHz ]]></Sign></Signs>
</Measure>
<Name ID="1291" Value="Processor clock speed" langid="1"/>
</Feature>
</ProductFeature>
```

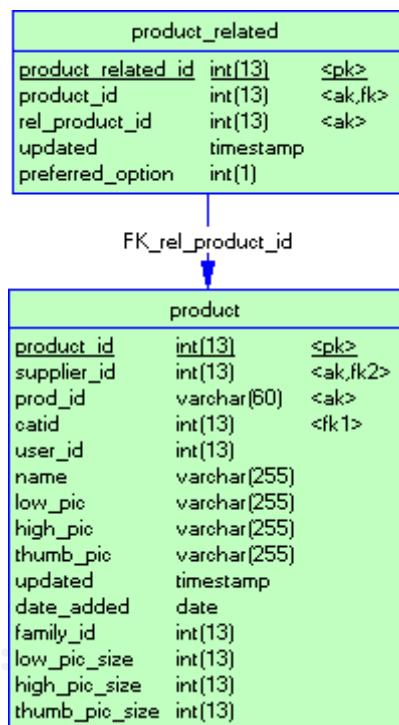


```
<SummaryDescription>
  <ShortSummaryDescription>Fujitsu AMILO D-7850 P4-3.2G HT 538 AMILO D, 3200 MHz, Intel Pentium 4, Intel Pentium IV, 533 MHz, SIS 648 + SIS 963L, 80 GB</ShortSummaryDescription>
  <LongSummaryDescription>Fujitsu AMILO D-7850 P4-3.2G HT 538, AMILO D. Processor: 3200 MHz, Intel Pentium 4, Intel Pentium IV, 533 MHz, SIS 648 + SIS 963L. Disk drive: 80 GB...</LongSummaryDescription>
</SummaryDescription>
```

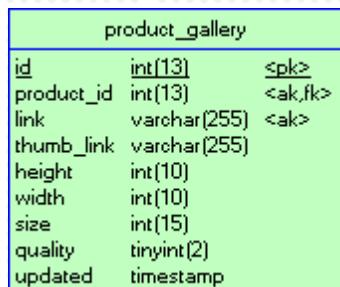
product_summary_description		
product_summary_description_id	int(13)	<pk>
product_id	int(13)	<ak>
langid	int(5)	<ak>
short_summary_description	text	
long_summary_description	text	
updated	timestamp	

```
<ProductRelated ID="0" Category_ID="153" Reversed="0" Preferred="0">
    <Product ID="1446916" Prod_id="AK818AW" ThumbPic="http://images2.icecat.biz/thumbs/1777706.jpg" Name="Compaq
dc5800 Small Form Factor PC">
        <Supplier ID="1" Name="HP"/>
    </Product>
</ProductRelated>
```

Note: If ProductRelated@ID = 0, it means, that this relation was generated dynamically, according to RelationsList.xml rules.



```
<ProductGallery>
    <ProductPicture ProductPicture_ID="78919"
PicWidth="109"
PicHeight="67"
Size="10280"
Pic="http://images.icecat.biz/img/gallery/1342044_3515.jpg"
ThumbPic="http://images.icecat.biz/img/gallery_thumbs/1342044_7401.jpg"/>
</ProductGallery>
```



```
<ProductMultimediaObject>
  <MultimediaObject URL="http://objects.Icecat.biz/objects/141648\_7817.swf
    Height="0"
    KeepAsURL="0"
    ContentType="application/x-shockwave-flash"
    langid="2"
    Width="0"
    Date="2008-11-20 19:46:26"
    Type="standard"
    Size="388978"
    MultimediaObject_ID="10"
    Description="Flash Demo"/>
</ProductMultimediaObject>
```

product_multimedia_object		
<code>id</code>	<code>int(13)</code>	<code><pk></code>
<code>product_id</code>	<code>int(13)</code>	<code><fk1,fk2,fk4></code>
<code>link</code>	<code>varchar(255)</code>	<code><fk4></code>
<code>short_descri</code>	<code>text</code>	
<code>langid</code>	<code>int(13)</code>	<code><fk2,fk3></code>
<code>size</code>	<code>int(15)</code>	
<code>updated</code>	<code>timestamp</code>	
<code>content_type</code>	<code>varchar(255)</code>	
<code>keep_as_url</code>	<code>int(1)</code>	
<code>type</code>	<code>enum('standard','presentation','movie')</code>	
<code>height</code>	<code>int(13)</code>	
<code>width</code>	<code>int(13)</code>	

5.5 How to use videos that are hosted by Icecat?

To use (download and/or play) video files from Icecat, two requirements should be met:

- the user (shopname) should be registered via Icecat.biz;
- the URL to stream the video should include the shopname of the Icecat user.

Below is an example of a URL to access a video via Icecat:

http://objects.icecat.biz/objects/3948536_9138.mp4?login=shopname

Without shopname in the URL request it is impossible to play or download a video that is hosted by Icecat.

Suggested video player platform: flash

5.6 Feature logos via Icecat XML

The feature logo functionality of Icecat provides you with logos and descriptions that are related to certain highlighted production specifications, such as compliance with EnergyStar, USB3.0, Bluetooth or the presence of Intel Inside technology.

Logos can be provided in JPG or PNG image formats and have a maximum resolution of up to 200 x 200 px. Textual descriptions of the feature logo are language specific.

Example product with feature logo xml included:

<http://data.icecat.biz/export/freexml/EN/13214266.xml>

Example xml:

```
<FeatureLogo ID="1" Feature_ID="2183" Value="Y" LogoPic="http://images.icecat.biz/img/feature_logo/1-7814.png"
Width="200" Height="200" Size="38243">
<Descriptions>
<Description ID="1" langid="1">
<![CDATA[
    Bluetooth is a proprietary open wireless technology standard for exchanging data over short distances (using short-wavelength radio transmissions in the ISM band from 2400–2480 MHz) from fixed and mobile devices, creating personal area networks (PANs) with high levels of security.
]]>
</Description>
</Descriptions>
</FeatureLogo>
```

The example xml above, shows the feature logo for Bluetooth technology which is supported in product 13214266.

The feature logo reference file can be found in the /refs/ directory:

<https://data.icecat.biz/export/freexml/refs/FeatureLogosList.xml.gz>

Explanation of used attributes:

FeatureLogo ID - IcecatID of the feature logo

Feature ID - IcecatID of the feature to which the logo is related

Value - the feature value in the product data-sheet for which the logo is shown

LogoPic - the logo image URL

Width and Height - the logo resolution (in pixels)

6. SQL set-up

6.1 Explanation of entities

Below on entity level, follows a short description of the entities used in our data model.

Category – table that holds the category structure information. Category names can be found in ‘vocabulary’, referenced via the ‘sid’ key
Category feature – link between feature and category
Category feature group – a group that holds a number of category features, to group them for display
Category keywords – category keywords that can be used for a search
Feature – holds the information about the features available for product description
Feature group – a generic features groups available in Icecat
Language – identifier of the language of a data element
Measure – units, e.g. meter, megabyte etc
Product – main information about the product
Product bundled – in case that a product is a distri bundle, info about components is here
Product description – language specific description
Product family – product lines families per supplier & category
Product series – product lines series per supplier, category & family
Product feature – product specs features are here. resolution of features/measures is via ->
category_feature -feature -> measure
Product gallery – some more product images
Product multimedia object – place for storing multimedia data, like swf, animated gifs, etc
Product related product cross-sell-relations or alternatives are stored here. The type of link can be determinened by categories. e.g. if the categories are the same -> link gives an “alternative”. Categories are different -> link is “option”.
Sid index – table for holding the autoincrement index for the vocabulary. needed only in case of issuing new records to the vocabulary
Supplier – manufacturers are stored in this table
Tex – vocabulary for large data elements
Tid index – index of tex table
Vocabulary – table for storing misc language dependent data

6.2 MySQL Set-Up (monolingual catalogue)

```
CREATE TABLE `category` (
  `catid` int(13) NOT NULL AUTO_INCREMENT,
  `ucatid` varchar(255) DEFAULT NULL,
  `pcatid` int(13) NOT NULL DEFAULT '1',
  `sid` int(13) NOT NULL DEFAULT '0',
  `tid` int(13) DEFAULT NULL,
  `searchable` int(3) NOT NULL DEFAULT '0',
  `low_pic` varchar(255) NOT NULL DEFAULT "",
```

```
`thumb_pic` varchar(255) DEFAULT "",  
 `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
 `last_published` int(14) DEFAULT '0',  
 `watched_top10` int(3) NOT NULL DEFAULT '0',  
 `visible` int(3) NOT NULL DEFAULT '0',  
 `ssid` int(13) NOT NULL,  
 PRIMARY KEY (`catid`),  
 UNIQUE KEY `ucatid`(`ucatid`),  
 KEY `pcatid`(`pcatid`),  
 KEY `catid`(`catid`,`sid`),  
 KEY `searchable_2`(`searchable`,`catid`),  
 KEY `sid_index`(`sid`),  
 KEY `visible`(`visible`,`catid`),  
 KEY `ssid`(`ssid`)  
) ENGINE=MyISAM;  
  
CREATE TABLE `category_feature` (  
 `category_feature_id` int(13) NOT NULL AUTO_INCREMENT,  
 `feature_id` int(13) NOT NULL DEFAULT '0',  
 `catid` int(13) NOT NULL DEFAULT '0',  
 `no` int(5) NOT NULL DEFAULT '0',  
 `searchable` int(3) NOT NULL DEFAULT '0',  
 `category_feature_group_id` int(13) NOT NULL DEFAULT '0',  
 `restricted_search_values` mediumtext,  
 `use_dropdown_input` char(3) DEFAULT "",  
 `mandatory` tinyint(2) DEFAULT '0',  
 `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
 PRIMARY KEY (`category_feature_id`),  
 UNIQUE KEY `feature_id`(`feature_id`,`catid`),  
 KEY `catid`(`catid`),  
 KEY `category_feature_id`(`category_feature_id`,`feature_id`)  
) ENGINE=MyISAM;
```

```
CREATE TABLE `category_feature_group` (
  `category_feature_group_id` int(13) NOT NULL AUTO_INCREMENT,
  `catid` int(13) NOT NULL DEFAULT '0',
  `feature_group_id` int(13) NOT NULL DEFAULT '0',
  `no` int(15) DEFAULT '0',
  PRIMARY KEY (`category_feature_group_id`),
  UNIQUE KEY `catid` (`catid`,`feature_group_id`)
) ENGINE=MyISAM;
```

```
CREATE TABLE `category_keywords` (
  `category_id` int(11) DEFAULT NULL,
  `langid` int(1) NOT NULL DEFAULT '0',
  `keywords` mediumtext,
  `id` int(11) NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`id`),
  UNIQUE KEY `langid` (`langid`,`category_id`),
  KEY `category_id` (`category_id`),
  FULLTEXT KEY `keywords` (`keywords`)
) ENGINE=MyISAM;
```

```
CREATE TABLE `feature` (
  `feature_id` int(13) NOT NULL AUTO_INCREMENT,
  `sid` int(13) NOT NULL DEFAULT '0',
  `tid` int(13) NOT NULL DEFAULT '0',
  `measure_id` int(13) NOT NULL DEFAULT '0',
  `type` varchar(60) NOT NULL DEFAULT '',
  `class` tinyint(3) NOT NULL DEFAULT '0',
  `limit_direction` tinyint(3) NOT NULL DEFAULT '0',
  `searchable` tinyint(3) NOT NULL DEFAULT '0',
  `restricted_values` mediumtext,
  `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `last_published` int(14) DEFAULT '0',
  PRIMARY KEY (`feature_id`),
```

```
KEY `tid` (`tid`),
KEY `sid` (`sid`)
) ENGINE=MyISAM;

CREATE TABLE `feature_group` (
`feature_group_id` int(13) NOT NULL AUTO_INCREMENT,
`sid` int(13) NOT NULL DEFAULT '0',
PRIMARY KEY (`feature_group_id`),
KEY `sid` (`sid`)
) ENGINE=MyISAM;

CREATE TABLE `language` (
`langid` int(3) NOT NULL AUTO_INCREMENT,
`sid` int(13) NOT NULL DEFAULT '0',
`code` varchar(32) NOT NULL DEFAULT '',
`short_code` varchar(5) NOT NULL DEFAULT '',
`published` char(1) NOT NULL DEFAULT 'N',
`backup_langid` int(3) DEFAULT NULL,
`updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
PRIMARY KEY (`langid`),
KEY `short_code` (`short_code`)
) ENGINE=MyISAM;

CREATE TABLE `measure` (
`measure_id` int(13) NOT NULL AUTO_INCREMENT,
`sid` int(13) NOT NULL DEFAULT '0',
`tid` int(13) NOT NULL DEFAULT '0',
`sign` varchar(255) DEFAULT NULL,
`updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
`last_published` int(14) DEFAULT '0',
`system_of_measurement` enum('metric','imperial') NOT NULL DEFAULT 'metric',
PRIMARY KEY (`measure_id`),
KEY `sid` (`sid`),
KEY `tid` (`tid`),
```

```
KEY `updated` (`updated`),
KEY `last_published` (`last_published`)
) ENGINE=MyISAM;

CREATE TABLE `measure_sign` (
  `measure_sign_id` int(13) NOT NULL AUTO_INCREMENT,
  `measure_id` int(13) NOT NULL DEFAULT '0',
  `langid` int(13) NOT NULL DEFAULT '0',
  `value` varchar(255) NOT NULL DEFAULT '',
  `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `last_published` timestamp NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`measure_sign_id`),
  UNIQUE KEY `measure_id` (`measure_id`,`langid`)
) ENGINE=MyISAM;

CREATE TABLE `product` (
  `product_id` int(13) NOT NULL AUTO_INCREMENT,
  `supplier_id` int(13) NOT NULL DEFAULT '0',
  `prod_id` varchar(60) NOT NULL DEFAULT '',
  `catid` int(13) NOT NULL DEFAULT '0',
  `user_id` int(13) NOT NULL DEFAULT '1',
  `launch_date` int(17) DEFAULT NULL,
  `obsolescence_date` int(17) DEFAULT NULL,
  `name` varchar(255) NOT NULL DEFAULT '',
  `low_pic` varchar(255) NOT NULL DEFAULT '',
  `high_pic` varchar(255) NOT NULL DEFAULT '',
  `publish` char(1) NOT NULL DEFAULT 'Y',
  `public` char(1) NOT NULL DEFAULT 'Y',
  `thumb_pic` varchar(255) DEFAULT NULL,
  `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `date_added` date NOT NULL DEFAULT '2004-11-01',
  `family_id` int(13) NOT NULL DEFAULT '0',
  `dname` varchar(255) NOT NULL DEFAULT ''
)
```

```
`topseller` varchar(255) NOT NULL DEFAULT "",  
 `low_pic_size` int(13) DEFAULT '0',  
 `high_pic_size` int(13) DEFAULT '0',  
 `thumb_pic_size` int(13) DEFAULT '0',  
 `high_pic_width` int(13) NOT NULL DEFAULT '0',  
 `high_pic_height` int(13) NOT NULL DEFAULT '0',  
 `low_pic_width` int(13) NOT NULL DEFAULT '0',  
 `low_pic_height` int(13) NOT NULL DEFAULT '0',  
 `high_pic_origin` varchar(255) NOT NULL DEFAULT "",  
 `series_id` int(17) NOT NULL DEFAULT '1',  
 `checked_by_supereeditor` tinyint(1) NOT NULL DEFAULT '0',  
 `medium_pic` varchar(255) NOT NULL,  
 `medium_pic_size` int(13) NOT NULL,  
 `medium_pic_width` int(13) NOT NULL,  
 `medium_pic_height` int(13) NOT NULL,  
 `high_pic_origin_size` int(13) NOT NULL,  
 PRIMARY KEY (`product_id`),  
 UNIQUE KEY `prod_id_2` (`prod_id`,`supplier_id`),  
 KEY `user_id` (`user_id`),  
 KEY `date_added` (`date_added`),  
 KEY `name` (`name`),  
 KEY `supplier_id_2` (`supplier_id`,`catid`),  
 KEY `publish` (`publish`,`public`),  
 KEY `catid` (`catid`,`updated`),  
 KEY `updated` (`updated`),  
 KEY `catid_2` (`catid`,`supplier_id`,`product_id`)  
) ENGINE=InnoDB;  
  
CREATE TABLE `product_description` (  
 `product_description_id` int(13) NOT NULL AUTO_INCREMENT,  
 `product_id` int(13) NOT NULL DEFAULT '0',  
 `langid` int(13) NOT NULL DEFAULT '0',  
 `short_desc` varchar(3000) NOT NULL DEFAULT "",
```

```
`long_desc` mediumtext NOT NULL,  
 `specs_url` varchar(255) NOT NULL DEFAULT "",  
 `support_url` varchar(255) NOT NULL DEFAULT "",  
 `official_url` text,  
 `warranty_info` mediumtext,  
 `option_field_1` mediumtext,  
 `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
 `pdf_url` varchar(255) NOT NULL DEFAULT "",  
 `option_field_2` mediumtext,  
 `pdf_size` int(13) DEFAULT '0',  
 `manual_pdf_url` varchar(255) NOT NULL DEFAULT "",  
 `manual_pdf_size` int(13) DEFAULT '0',  
 `pdf_url_origin` text,  
 `manual_pdf_url_origin` text,  
 `pdf_updated` int(13) NOT NULL DEFAULT '0',  
 `manual_pdf_updated` int(13) NOT NULL DEFAULT '0',  
 PRIMARY KEY (`product_description_id`),  
 UNIQUE KEY `product_id` (`product_id`,`langid`),  
 KEY `updated` (`updated`)  
) ENGINE=InnoDB;  
  
CREATE TABLE `product_family` (  
 `family_id` int(17) NOT NULL AUTO_INCREMENT,  
 `parent_family_id` int(17) NOT NULL DEFAULT '1',  
 `supplier_id` int(17) NOT NULL DEFAULT '0',  
 `sid` int(13) NOT NULL DEFAULT '0',  
 `tid` int(13) NOT NULL DEFAULT '0',  
 `low_pic` varchar(255) DEFAULT NULL,  
 `thumb_pic` varchar(255) DEFAULT NULL,  
 `catid` int(13) NOT NULL DEFAULT '0',  
 `data_source_id` int(13) NOT NULL DEFAULT '0',  
 `symbol` varchar(120) NOT NULL DEFAULT "",
```

```
'updated' timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
PRIMARY KEY ('family_id'),  
KEY 'supplier_id_3' ('supplier_id','sid'),  
KEY 'sid' ('sid','supplier_id'),  
KEY 'data_source_id' ('data_source_id'),  
KEY 'symbol' ('symbol')  
) ENGINE=MyISAM;  
  
CREATE TABLE `product_series` (  
    `series_id` int(17) NOT NULL AUTO_INCREMENT,  
    `sid` int(13) NOT NULL,  
    `tid` int(13) NOT NULL,  
    `supplier_id` int(17) NOT NULL,  
    `catid` int(13) NOT NULL,  
    `family_id` int(17) NOT NULL,  
    PRIMARY KEY (`series_id`),  
    KEY `sid` ('sid'),  
    KEY `supplier_id` ('supplier_id'),  
    KEY `family_id` ('family_id')  
) ENGINE=MyISAM;  
  
CREATE TABLE `product_feature` (  
    `product_feature_id` int(13) NOT NULL AUTO_INCREMENT,  
    `product_id` int(13) NOT NULL DEFAULT '0',  
    `category_feature_id` mediumint(8) unsigned NOT NULL,  
    `value` varchar(20000) NOT NULL DEFAULT '',  
    `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
    PRIMARY KEY (`product_feature_id`),  
    UNIQUE KEY `category_feature_id_2` ('category_feature_id','product_id'),  
    KEY `product_id` ('product_id'),  
    KEY `value` ('value'(255)),  
    KEY `updated` ('updated'),  
    KEY `category_feature_id` ('category_feature_id','value'(250))
```

```
) ENGINE=InnoDB;

CREATE TABLE `product_feature_local` (
  `product_feature_local_id` int(13) NOT NULL AUTO_INCREMENT,
  `product_id` int(13) NOT NULL DEFAULT '0',
  `category_feature_id` int(13) NOT NULL DEFAULT '0',
  `value` varchar(15000) NOT NULL DEFAULT '',
  `langid` int(5) NOT NULL DEFAULT '0',
  `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`product_feature_local_id`),
  UNIQUE KEY `category_feature_id` (`category_feature_id`,`product_id`,`langid`),
  KEY `product_id` (`product_id`,`langid`),
  KEY `langid` (`langid`),
  KEY `value` (`value`(250))
) ENGINE=InnoDB;

CREATE TABLE `product_gallery` (
  `id` int(13) NOT NULL AUTO_INCREMENT,
  `product_id` int(13) NOT NULL DEFAULT '0',
  `link` varchar(255) NOT NULL DEFAULT '',
  `thumb_link` varchar(255) NOT NULL DEFAULT '',
  `height` int(10) NOT NULL DEFAULT '0',
  `width` int(10) NOT NULL DEFAULT '0',
  `size` int(15) NOT NULL DEFAULT '0',
  `quality` tinyint(2) DEFAULT '0',
  `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `thumb_size` int(15) NOT NULL DEFAULT '0',
  `link_origin` varchar(255) NOT NULL DEFAULT '',
  `low_link` varchar(255) NOT NULL,
  `medium_link` varchar(255) NOT NULL,
  `low_height` int(13) NOT NULL,
  `medium_height` int(13) NOT NULL,
  `low_width` int(13) NOT NULL,
```

```
`medium_width` int(13) NOT NULL,  
 `low_size` int(13) NOT NULL,  
 `medium_size` int(13) NOT NULL,  
 `size_origin` int(13) NOT NULL,  
 `langid` int(3) NOT NULL DEFAULT '0',  
 PRIMARY KEY (`id`),  
 UNIQUE KEY `product_id_2` (`product_id`,`link`),  
 KEY `updated` (`updated`),  
 KEY `product_id` (`product_id`,`langid`)  
) ENGINE=MyISAM;  
  
CREATE TABLE `product_multimedia_object` (  
 `id` int(13) NOT NULL AUTO_INCREMENT,  
 `product_id` int(13) NOT NULL DEFAULT '0',  
 `link` varchar(255) NOT NULL DEFAULT "",  
 `short_descr` mediumtext NOT NULL,  
 `langid` int(13) NOT NULL DEFAULT '0',  
 `size` int(15) NOT NULL DEFAULT '0',  
 `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
 `content_type` varchar(255) NOT NULL DEFAULT "",  
 `keep_as_url` int(1) NOT NULL DEFAULT '0',  
 `type` varchar(255) NOT NULL DEFAULT 'standard',  
 `height` int(13) NOT NULL DEFAULT '0',  
 `width` int(13) NOT NULL DEFAULT '0',  
 `data_source_id` int(13) NOT NULL DEFAULT '0',  
 `link_origin` varchar(255) NOT NULL DEFAULT "",  
 PRIMARY KEY (`id`),  
 UNIQUE KEY `product_id_3` (`product_id`,`langid`,`link`(200)),  
 KEY `product_id` (`product_id`,`updated`),  
 KEY `product_id_2` (`product_id`,`langid`),  
 KEY `data_source_id` (`data_source_id`,`product_id`),  
 KEY `type` (`type`),
```

```
KEY `updated` (`updated`)
) ENGINE=MyISAM;

CREATE TABLE `product_related` (
  `product_related_id` int(13) NOT NULL AUTO_INCREMENT,
  `product_id` int(13) NOT NULL DEFAULT '0',
  `rel_product_id` int(13) NOT NULL DEFAULT '0',
  `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `preferred_option` tinyint(1) NOT NULL DEFAULT '0',
  `data_source_id` int(13) NOT NULL DEFAULT '0',
  `order` smallint(5) unsigned NOT NULL DEFAULT '65535',
  PRIMARY KEY (`product_related_id`),
  UNIQUE KEY `product_id` (`product_id`,`rel_product_id`),
  KEY `rel_product_id` (`rel_product_id`),
  KEY `data_source_id` (`data_source_id`,`product_id`,`rel_product_id`)
) ENGINE=InnoDB;

CREATE TABLE `product_review` (
  `product_review_id` int(13) NOT NULL AUTO_INCREMENT,
  `product_id` int(13) NOT NULL DEFAULT '0',
  `langid` int(13) NOT NULL DEFAULT '0',
  `review_group` varchar(60) NOT NULL DEFAULT '',
  `review_code` varchar(60) NOT NULL DEFAULT '',
  `review_id` int(13) NOT NULL DEFAULT '0',
  `score` int(13) NOT NULL DEFAULT '0',
  `url` varchar(255) NOT NULL DEFAULT '',
  `logo_url` varchar(255) NOT NULL DEFAULT '',
  `value` blob,
  `value_good` blob,
  `value_bad` blob,
  `postscriptum` blob,
  `review_award_name` varchar(120) NOT NULL DEFAULT '',
  `high_review_award_url` varchar(255) NOT NULL DEFAULT ''
```

```
`low_review_award_url` varchar(255) NOT NULL DEFAULT "",  
`date_added` date DEFAULT NULL,  
`updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,  
PRIMARY KEY (`product_review_id`),  
UNIQUE KEY `product_id` (`product_id`,`review_id`,`langid`),  
KEY `date_added` (`date_added`),  
KEY `review_group` (`review_group`,`product_id`),  
KEY `updated` (`updated`)  
) ENGINE=MyISAM;  
  
CREATE TABLE `sid_index` (  
    `sid` int(13) NOT NULL AUTO_INCREMENT,  
    `dummy` int(1) DEFAULT NULL,  
PRIMARY KEY (`sid`)  
) ENGINE=MyISAM;  
  
CREATE TABLE `supplier` (  
    `supplier_id` int(13) NOT NULL AUTO_INCREMENT,  
    `user_id` int(13) NOT NULL DEFAULT '1',  
    `name` varchar(255) NOT NULL DEFAULT "",  
    `low_pic` varchar(255) DEFAULT NULL,  
    `thumb_pic` varchar(255) DEFAULT NULL,  
    `acknowledge` char(1) NOT NULL DEFAULT 'N',  
    `is_sponsor` char(1) NOT NULL DEFAULT 'N',  
    `public_login` varchar(80) DEFAULT "",  
    `public_password` varchar(80) DEFAULT "",  
    `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
    `last_published` int(14) DEFAULT '0',  
    `ftp_homedir` varchar(255) DEFAULT NULL,  
    `template` mediumtext,  
    `folder_name` varchar(255) NOT NULL DEFAULT "",  
    `suppress_offers` char(1) NOT NULL DEFAULT 'N',  
    `last_name` varchar(255) NOT NULL DEFAULT "",
```

```
`prod_id_regex` text,  
 `has_vendor_index` tinyint(1) NOT NULL DEFAULT '0',  
 `hide_products` tinyint(1) NOT NULL DEFAULT '0',  
 PRIMARY KEY (`supplier_id`),  
 UNIQUE KEY `name` ('name'),  
 KEY `is_sponsor` (`is_sponsor`),  
 KEY `public_login` (`public_login`),  
 KEY `folder_name` ('folder_name'),  
 KEY `updated` ('updated'),  
 FULLTEXT KEY `fulltext_name` ('name')  
) ENGINE=MyISAM;  
  
CREATE TABLE `tex` (  
 `tex_id` int(13) NOT NULL AUTO_INCREMENT,  
 `tid` int(13) NOT NULL DEFAULT '0',  
 `langid` int(3) NOT NULL DEFAULT '0',  
 `value` mediumtext,  
 PRIMARY KEY (`tex_id`),  
 UNIQUE KEY `tid` ('tid','langid'),  
 KEY `langid` ('langid')  
) ENGINE=MyISAM;  
  
CREATE TABLE `tid_index` (  
 `tid` int(13) NOT NULL AUTO_INCREMENT,  
 `dummy` int(1) DEFAULT NULL,  
 PRIMARY KEY (`tid`)  
) ENGINE=MyISAM;  
  
CREATE TABLE `vocabulary` (  
 `record_id` int(13) NOT NULL AUTO_INCREMENT,  
 `sid` int(13) NOT NULL DEFAULT '0',  
 `langid` int(3) NOT NULL DEFAULT '0',  
 `value` varchar(255) DEFAULT NULL,  
 `updated` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
```

```
PRIMARY KEY (`record_id`),  
UNIQUE KEY `sid_2` (`sid`,`langid`),  
KEY `langid` (`langid`),  
KEY `updated` (`updated`)  
) ENGINE=MyISAM;
```

