

upgradebox.eu

Application Programming Interface (API)

(upgbAPI)

Documentation

Version: 1.3

Doc-Version: 1.3
Doc-Date: 2015-07-02

First Date: 2008-07-05
Author: Thorsten Preuss <tp@upgradebox.eu>

This documentation and the programming samples are to be treated confidential! The upgbAPI is intended to be used from within your internet enabled applications only. You are not allowed to store the information provided by the upgbAPI in your own databases except for caching purposes.

Table of Contents

1.	General tasks of the upgbAPI.....	3
2.	Communication with the upgbAPI.....	4
2.1	Test Environment.....	6
3.	Encryption and authorization.....	7
4.	Working modes.....	8
4.1	Model Search Mode.....	10
4.2	Free Text Search Mode.....	13
4.3	Product Feature Search Mode.....	15
4.4	Article Functions.....	16
4.5	Subuser Maintainance Mode.....	18
5.	Examples.....	20
6.	Version history.....	25

1. General tasks of the upgbAPI

The upgbAPI provides access to the upgradebox.eu configurator databases using a XML-based SSL-encrypted HTTP POST interface. All search modes provided by the browser based interface are available using the upgbAPI.

The following search modes are available:

- search for manufacturer / model type / model
- free text wildcard search for models, article numbers, article name and OEM numbers
- search for products with specific features

Alike the browser based interface, the model manufacturer (e.g. Sun, HP/Compaq, Dell, ...) and article manufacturer (e.g. Kingston Technology, Origin Storage,...) limitations for your customer account apply and your article numbers are presented, if these are stored within the upgradebox.eu. You can also choose your desired output language from the available languages (7 languages are available by the time of writing, further languages are available soon. Users of the upgbAPI will be informed of new language options by email).

Access to the upgbAPI is unrestricted. However, in the future the number of api calls might be restricted to certain frequencies (e.g. 10/min, 1000/day).

2. Communication with the upgbAPI

The communication with the upgbAPI is established by HTTP POST to the SSL-encrypted URL

`https://api.upgradebox.net/[VERSION]/`

The POST data should contain a XML document including

- the XML header with the content type definition and the charset definition
- the authorization information (namely your custID, subID and your password)
- your own session information (e.g. the name of the user of your application, the serial number), which are sent back if the answer requested is a XML document (up to 5 variables named *custvar1 .. custvar5* can be submitted)
- the request parameters
 - request type
 - request result language
 - request type specific parameters

```
<?xml version="1.0" encoding="UTF-8"?>
<upgradeboxrequest>
  <authorization>
    <userid>19823019</userid>
    <subid>0000</subid>
    <pass>SAMPLEPASS</pass>
  </authorization>
  <customvariables>
    <custvar1>REQ0001</custvar1>
    <custvar2>[CUST#]</custvar2>
    <custvar3>SESS238zr7fh485679382490</custvar3>
  </customvariables>
  <request>
    <requesttype>ModelManufacturers</requesttype>
    <requestlanguage>EN</requestlanguage>
    <requestformat>XML</requestformat>
  </request>
</upgradeboxrequest>
```

(1) XML Sample: Simple Request

The HTTP answer consists, depending on the requested format, of

- a XML document containing
 - a XML header with the content type definition and the charset definition
 - your session information, if provided within the request
 - the search results in the desired language

```
<?xml version="1.0" encoding="UTF-8"?>
<upgradeboxanswer>
  <customvariables>
    <custvar1>REQ0001</custvar1>
    <custvar2>CUST#</custvar2>
    <custvar3>SESS238zr7fh485679382490</custvar3>
  </customvariables>
  <requested>
    <requesttype>GetModelManufacturers</requesttype>
    <resultlanguage>EN</resultlanguage>
    <resultformat>XML</resultformat>
  </requested>
  <resulttype>List</resulttype>
  <result>
    <manufacturer>ABIT</manufacturer>
```

```
<manufacturer>Acer</manufacturer>
....
<manufacturer>Zenith</manufacturer>
</result>
</upgradeboxanswer>
```

(2) *XML Sample: Simple Answer*

- a plain text document with the search results (lines are separated by NEWLINE characters (`\n`; `chr(10)`), fields are separated by tabulators (`\t`; `chr(9)`) including the field names as the first row of the results

```
Manufacturers
ABIT
Acer
...
Zenith
```

(3) *Text Sample: Simple Answer*

- a PDF document
- a JPEG or PNG picture

The server answer will be send back with the correct HTTP header content type definitions so that the result can be displayed directly by HTTP aware applications.

2.1 Test Environment

There is a simple test environment available

[https://api.upgradebox.net/\[VERSION\]/test.php](https://api.upgradebox.net/[VERSION]/test.php)

where the XML requests can be edited by hand and the results are displayed with the browsers XML style sheet.

You can copy the examples from chapter 5 into the textarea to see the live results the upgbAPI produces. Except for the SubUser Mode (which is not allowed to be used when you login with a subid other than 0000) you can test all functions and compare the results with the output of the upgradebox running under the standard URL <http://www.upgradebox.eu> (the userid 12873 actually is the id the common upgradebox uses).

3. Encryption and authorization

The communication with the upgbAPI server is 128-bit SSL-encrypted and is authenticated by an official certificate.

For authorization, the userID, the subID and the password has to be provided within the XML request. The userID for the initial subID 0000 and the password is provided by the upgradebox.eu support team when setting up your customer account. With a special upgbAPI call you can set up and also change/delete additional subusers with individual passwords linked to your customer setup for the usage from within your applications (which e.g. can then be given to your customers without providing the subID 0000 password). With a upgbAPI request you can request a usage statistic for a single subID for a time period defined by the start and end date within the request.

Your UserID: _____

Your SubID: 0000

Your Subuser Name: Main Account

Your Password: _____

Additional Subuser ID's

SubID	Subuser Name (for reports)	Your remarks	Password
0001			
0002			
0003			
0004			
0005			
0006			
0007			
0008			
0009			
0010			

4. Working modes

The upgbAPI provides 3 different search modes, some article functions and a few extra functions for the maintenance of your subuser accounts. The requests are send as XML documents, the answer can be returned in one of the following formats, while not all formats are valid for all request types (see *Table 1*).

Mode	Request type	XML	HTML	TEXT	IMG	PDF
Model Search	ModelManufacturers	X		X		
	ModelTypes	X		X		
	ModelTypesList	X		X		
	ModelListManufacturer	X		X		
	ModelListManufacturerType	X		X		
	ModelArticles	X		X		
	ModelInfo	X		X		
	ModelPicture				X	
	ModelDatasheet					X
	ModelManual					X
Free Text	FreeTextModels	X		X		
	FreeTextArticleReference	X		X		
	FreeTextArticleName	X		X		
	FreeTextArticleOEM	X		X		
	FreeTextArticle	X		X		
Feature	FeatureGroups	X		X		
	FeatureTypes	X		X		
	FeatureTypeValues	X		X		
	FeatureArticles	X		X		
Article	ArticlePicture				X	
	ArticleThumb				X	
	ArticleModels	X		X		
	ArticleAccessories	X		X		
	ArticleDetails	X		X		X
Subuser Maintenance	SubUserList	X		X		
	SubUserAdd	X				
	SubUserChange	X				
	SubUserDelete	X				
	SubUserStatistics	X		X		

Table 1: Request Type / Answer Type Matrix

The desired result language can be included into the XML; if omitted, the result will be returned in the default language English. For a list of available languages see *Table 2*.

Language	Code
Czech	CZ
Danish	DK
Dutch	NL
English (default)	EN
French	FR
German	DE
Hungarian	HU
Italian	IT
Polish	PL
Portuguese	PT
Slovakian	SL
Spanish	ES
Swedish	SK
Ukrainian	UA

Table 2: Language Codes

4.1 Model Search Mode

Command	ModelManufacturers
Function	Get a list of available model manufacturers. Optionally the list of manufacturers can be reduced to those for which articles of the model type <i>requirehasmodeltype</i> are present.
Parameters	-
Option	<requirehasmodeltype>[MODEL TYPE SYSTEM NAME]</requirehasmodeltype>
Results	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> ...

Command	ModelTypes
Function	List available model type categories for the given manufacturer
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer>
Results	<modeltype> <modeltypesystemname>[MODEL TYPE SYSTEM NAME]</modeltypesystemname> <modeltypedisplay>[MODEL TYPE DISPLAY NAME]</modeltypedisplay> </modeltype> ...

Command	ModelTypesList
Function	List generally available model types.
Parameters	-
Results	<modeltype> <modeltypesystemname>[MODEL TYPE SYSTEM NAME]</modeltypesystemname> <modeltypedisplay>[MODEL TYPE DISPLAY NAME]</modeltypedisplay> </modeltype> ...

Command	ModelListManufacturer
Function	List models for the given manufacturer and model type
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer>
Results	<model> <modeltypesystemname>[MODEL TYPE SYSTEM NAME]</modeltypesystemname> <modeltypedisplay>[MODEL TYPE DISPLAY]</modeltypedisplay> <modelname>[MODEL NAME]</modelname> </model> ...

Command	ModelListManufacturerType
Function	List models for the given manufacturer and model type
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modeltypesystemname>[MODEL TYPE SYSTEM NAME]</modeltypesystemname>
Results	<modelname>[MODEL NAME]</modelname> ...

Command	ModelArticles
Function	List available articles for the given manufacturer, model type and model
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modelName>[MODEL NAME]</modelName>
Results	<article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articlecompattype>[ARTICLE COMPATIBILITY TYPE]</articlecompattype> </article> ...

Command	ModelInfo
Function	List available model information for the given manufacturer, model type and model. If a picture/thumb is available, haspicture/hasthumb returns 1, otherwise 0. If a datasheet or a manual is available as PDF document, hasdatasheet/hasmanual returns 1 and the value datasheetlanguage/manuallanguage indicates, whether the PDF is available in the requested language (native) or only in an English version (english).
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modelName>[MODEL NAME]</modelName>
Results	<modelinfo> <oemsysid>[OEM SYS IDS]</oemsysid> <displaysize>[DISPLAY SIZE]</displaysize> <sysean>[SYSEAN]</sysean> <asin>[ASIN]</asin> <custmodelcode>[CUSTOMER SPECIFIC MODEL CODE]</custmodelcode> <serialnumber>[SERIAL NUMBERS]</serialnumber> <busarchitecture>[BUSARCHITECTURES]</busarchitecture> <cpuchipsets> <cpuchipset>[CPUCHIPSET]</cpuchipset> ... <cpuchipsets> <standardmemory>[STANDARDMEMORY]</standardmemory> <expansion>[EXPANSION]</expansion> <maximummemory>[MAXIMUMMEMORY]</maximummemory> <bankschemata> <bankschema>[BANKSCHEMA]</bankschema> ... <bankschemata> <haspicture>[0/1]</haspicture> <hasdatasheet>[0/1]</hasdatasheet> <datasheetlanguage>[native/english]</datasheetlanguage> <hasmanual>[0/1]</hasmanual> <manuallanguage>[native/english]</manuallanguage> <osmemorystandard>[OSMEMORYSTANDARD]</osmemorystandard> <osmemorymaximum>[OSMEMORYMAXIMUM]</osmemorymaximum> <osmemorysockets>[OSMEMORYSOCKETS]</osmemorysockets> <osmemorynotes>[OSMEMORYNOTES]</osmemorynotes> </modelinfo>

Command	ModelPicture
Function	Get the JPEG or PNG picture for the given article, returns a transparent GIF in case of an error (e.g. when the required picture does not exist)
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modelName>[MODEL NAME]</modelName>
Results	[RAW PICTURE DATA STREAM]

Command	ModelDatasheet
Function	Get the PDF of the datasheet for the given model, returns an error message in case of an error (e.g. when the required PDF document does not exist)
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modelName>[MODEL NAME]</modelName>
Results	[PDF DATA STREAM]

Command	ModelManual
Function	Get the PDF of the manual for the given model, returns an error message in case of an error (e.g. when the required PDF document does not exist)
Parameters	<modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modelName>[MODEL NAME]</modelName>
Results	[PICTURE DATA STREAM]

4.2 Free Text Search Mode

Command	FreeTextModels
Function	Search for models including the given search term. Optionally the list of models can be reduced to those of the given model type <i>restrictmodeltype</i> .
Parameters	<searchterm>[SEARCH TERM]</searchterm>
Option	<restrictmodeltype>[MODEL TYPE SYSTEM NAME]</ restrictmodeltype>
Results	<pre> <model> <modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modeltypesystemname>[MODEL TYPE SYSTEM NAME]</modeltypesystemname> <modeltypedisplay>[MODEL TYPE DISPLAY]</modeltypedisplay> <modelname>[MODEL NAME]</modelname> </model> ... </pre>

Command	FreeTextArticleReference
Function	Search for articles with reference numbers (original manufacturer numbers) including the given search term
Parameters	<searchterm>[SEARCH TERM]</searchterm>
Results	<pre> <article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articleoem>[ARTICLE OEM NUMBERS]</articleoem> </article> ... </pre>

Command	FreeTextArticleName
Function	Search for articles with names including the given search term
Parameters	<searchterm>[SEARCH TERM]</searchterm>
Results	<pre> <article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articleoem>[ARTICLE OEM NUMBERS]</articleoem> </article> ... </pre>

Command	FreeTextArticleOEM
Function	Search for articles with oem numbers including the given search term
Parameters	<searchterm>[SEARCH TERM]</searchterm>
Results	<pre> <article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articleoem>[ARTICLE OEM NUMBERS]</articleoem> </article> ... </pre>

Command	FreeTextArticle
Function	Search for articles where either of the search fields (reference number, name or oem numbers) includes the given search term
Parameters	<searchterm>[SEARCH TERM]</searchterm>
Results	<pre> <article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articleoem>[ARTICLE OEM NUMBERS]</articleoem> </article> ... </pre>

4.3 Product Feature Search Mode

Command	FeatureGroups
Function	Get a list of available product feature groups (e.g. Bags and cases, Display Protection)
Parameters	-
Results	<pre><featuregroup> <featuregroupssystemname>[FEATURE GROUP SYSTEM NAME]</featuregroupssystemname> <featuregroupdisplay>[FEATURE GROUP DISPLAY]</featuregroupdisplay> </featuregroup> ...</pre>

Command	FeatureTypes
Function	Get a list of feature types for the given feature group (e.g. 1. Displaysize, 2. Material, 3. Colour and 4. Type for Bags and cases)
Parameters	<featuregroupssystemname>[FEATURE GROUP SYSTEM NAME]</featuregroupssystemname>
Results	<pre><featuretype> <featuretypesystemname>[FEATURE TYPE SYSTEM NAME]</featuretypesystemname> <featuretypedisplay>[FEATURE TYPE DISPLAY]</featuretypedisplay> </featuretype> ...</pre>

Command	FeatureTypeValues
Function	Get a list of possible values for the given feature group and feature type (e.g. Backpac, Frontloader... for 4. Type for Bags and cases)
Parameters	<pre><featuregroupssystemname>[FEATURE GROUP SYSTEM NAME]</featuregroupssystemname> <featuretypesystemname>[FEATURE TYPE SYSTEM NAME]</featuretypesystemname></pre>
Results	<pre><featurevalue>[FEATURE VALUE]</featurevalue> ...</pre>

Command	FeatureArticles
Function	List available articles for a given feature group with the chosen feature type values
Parameters	<featuregroupssystemname>[FEATURE GROUP SYSTEM NAME]</featuregroupssystemname>
Optional	<pre><featuretype0>[FEATURE TYPE SYSTEM NAME]</featuretype0> <featurevalue0>[FEATURE VALUE SYSTEM NAME]</featurevalue0> ... <featuretype[n]>[FEATURE TYPE SYSTEM NAME]</featuretype[n]> <featurevalue[n]>[FEATURE VALUE SYSTEM NAME]</featurevalue[n]></pre>
Results	<pre><article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> </article> ...</pre>

4.4 Article Functions

Command	ArticlePicture
Function	Get the JPEG picture with the given ID for the given article, returns a transparent GIF in case of an error (e.g. when the required picture does not exist)
Parameters	<articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference>
Optional	<articlepictureid>[PICTURE NUMBER]</articlepictureid>
Results	[RAW PICTURE DATA STREAM]

Command	ArticleThumb
Function	Get the JPEG thumbnail picture for the given article, returns a transparent GIF in case of an error (e.g. when the required picture does not exist)
Parameters	<articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference>
Results	[RAW PICTURE DATA STREAM]

Command	ArticleModels
Function	Get a list of models the given article is compatible to
Parameters	<articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference>
Results	<model> <modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modeltypesystemname>[MODEL TYPE SYSTEM NAME]</modeltypesystemname> <modeltypedisplay>[MODEL TYPE DISPLAY]</modeltypedisplay> <modelname>[MODEL NAME]</modelname> </model> ...

Command	ArticleAccessories
Function	Get a list of articles marked as accessories for the given article number
Parameters	<articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference>
Results	<article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articleoem>[ARTICLE OEM NUMBERS]</articleoem> </article> ...

Command	ArticleDetails
Function	Get article details fo the given article reference number
Parameters	<pre><articlemanufacturer>[MANUFACTURER]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference></pre>
Results	<pre><article> <articlegroup>[ARTICLE GROUP]</articlegroup> <articlesubgroup>[ARTICLE SUB GROUP]</articlesubgroup> <articlemanufacturer>[ARTICLE MANUFACTURER]</articlemanufacturer> <articleordernumber>[ARTICLE ORDER 1NUMBER]</articleordernumber> <articlereference>[ARTICLE REFERENCE NUMBER]</articlereference> <articlename>[ARTICLE NAME]</articlename> <articleean>[ARTICLE EAN NUMBER]</articleean> <articleoem>[ARTICLE OEM NUMBERS]</articleoem> <articledescription>[ARTICLE DESCRIPTION]</articledescription> <articlewarranty>[ARTICLE WARRANTY]</articlewarranty> <articleremarks>[ARTICLE REMARKS]</articleremarks> <articlepictures>[NUMBER OF PICTURES [0,1,2]]</articlepictures> <articlecompatiblemodels>[NUMBER OF COMPATIBLE MODELS]</articlecompatiblemodels> <articleaccessories>[NUMBER OF ACCESSORIES]</articleaccessories> </article> </article></pre>

4.5 Subuser Maintenance Mode

Command	SubUserList
Function	Return a list of available subusers
Parameters	-
Results	<pre> <subuser> <subuserid>[SUBUSER ID]</subuserid> <subusername>[SUBUSER NAME]</subusername> <subuserstatus>[SUBUSER STATUS]</subuserstatus > <subuserpassword>[SUBUSER PASSWORD]</subuserpassword> <subuserremarks>[SUBUSER REMARKS]</subuserremarks> </subuser> ... </pre>

Command	SubUserAdd
Function	Create a new subuser
Parameters	<pre> <subuserpassword>[SUBUSER PASSWORD]</subuserpassword> <subusername>[SUBUSER NAME]</subusername> </pre>
Optional	<pre> <subuserremarks>[SUBUSER REMARKS]</subuserremarks> </pre>
Results	<pre> <subuserid>[SUBUSER ID]</subuserid> </pre>

Command	SubUserChange
Function	Change a subuser
Parameters	<pre> <subuserid>[SUBUSER ID]</subuseid> </pre>
Optional	<pre> <subusername>[SUBUSER NAME]</subusername> <subuserstatus>[SUBUSER STATUS [active or inactive]]</subuserstatus > <subuserpassword>[SUBUSER PASSWORD]</subuserpassword> <subuserremarks>[SUBUSER REMARKS]</subuserremarks> </pre>
Results	<pre> <subuserschanged>[NUMBER OF CHANGED SUBUSERS]</subuserschanged> </pre>

Command	SubUserDelete
Function	Delete a subuser
Parameters	<pre> <subuserid>[SUBUSER ID]</subuseid> </pre>
Results	<pre> <subusersdeleted>[NUMBER OF DELETED SUBUSERS]</subusersdeleted> </pre>

Command	SubUserStatistics
Function	Get the usage statistics for the given subuser ordered by date and time. Optionally you can set the start and/or the end date for the report.
Parameters	<subuserid>[SUBUSER ID]</subuseid>
Optional	<date_start>[YYYY-MM-DD]</date_start> <date_end>[YYYY-MM-DD]</date_end>
Results	<pre> <apicall> <errorcode>[ERROR CODE]</errorcode> <errorstring>[ERROR STRING]</errorstring> <date>[YYYY-MM-DD]</date> <time>[HH:mm:ss]</time> <host>[REMOTE HOST IP AND NAME]</host> <type>[REQUEST TYPE]</type> <language>[REQUEST LANGUAGE]</language> <format>[REQUEST FORMAT]</format> <modelmanufacturer>[MODEL MANUFACTURER NAME]</modelmanufacturer> <modelname>[MODEL NAME]</modelname> <articlemanufacturer>[ARTICLE MANUFACTURER NAME]</articlemanufacturer> <articlereference>[ARTICLE REFERENCE]</articlereference> </apicall> ... </pre>

5. Examples

The following examples can be used directly from within the testing environment (see 2.1).

Function	Basic query for all available model manufacturers
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ModelManufacturers</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> </request> </upgradeboxrequest></pre>
Function	After getting the correct manufacturer name the next request results in a list of available models for the given model manufacturer (e.g. DELL)
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ModelListManufacturer</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> <modelmanufacturer>DELL</modelmanufacturer> </request> </upgradeboxrequest></pre>
Function	With the correct model name the next request returns a complete list of available articles for the chosen model
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ModelArticles</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> <modelmanufacturer>DELL</modelmanufacturer> <modelName>Latitude D600</modelName> </request> </upgradeboxrequest></pre>

Function	Request a list of available feature groups
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>FeatureGroups</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> </request> </upgradeboxrequest></pre>
Function	Request available feature types for "Bags and Cases"
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables> <custvar1>REQ0001</custvar1> <custvar2>[CUST#]</custvar2> <custvar3>SESS238zr7fh485679382490</custvar3> </customvariables> <request> <requesttype>FeatureTypes</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> <featuregroupssystemname>Bags and cases</featuregroupssystemname> </request> </upgradeboxrequest></pre>
Function	Request available values for "3. Colour"
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>FeatureTypeValues</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> <featuregroupssystemname>Bags and cases</featuregroupssystemname> <featuretypesystemname>3. Colour</featuretypesystemname> </request> </upgradeboxrequest></pre>

Function	Request all articles of the article group "Bags and Cases" where "3. Colour" is "Black"
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>FeatureArticles</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> <featuregroupssystemname>Bags and cases</featuregroupssystemname> <featuretype0>3. Colour</featuretype0> <featurevalue0>Black</featurevalue0> </request> </upgradeboxrequest></pre>

Function	Get the second picture for the article "Targus Bags" "TCM005EU"
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ArticlePicture</requesttype> <requestlanguage>EN</requestlanguage> <requestformat>IMG</requestformat> <articlemanufacturer>Targus Bags</articlemanufacturer> <articlereference>TCM005EU</articlereference> <articlepictureid>2</articlepictureid> </request> </upgradeboxrequest></pre>

Function	Request for all accessories for the "Avocent" "16SV1000-201" KVM switch
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ArticleAccessories</requesttype> <requestlanguage>EN</requestlanguage> <requestformat>XML</requestformat> <articlemanufacturer>Avocent</articlemanufacturer> <articlereference>16SV1000-201</articlereference> </request> </upgradeboxrequest></pre>

Function	Get the french PDF for "Targus Bags" article "TCM005EU"
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ArticleDetails</requesttype> <requestlanguage>FR</requestlanguage> <requestformat>PDF</requestformat> <articlemanufacturer>Targus Bags</articlemanufacturer> <articlereference>TCM005EU</articlereference> </request> </upgradeboxrequest></pre>

Function	Get the german XML for "Targus Bags" article "TCM005EU"
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0001</subid> <pass>testpassword</pass> </authorization> <customvariables /> <request> <requesttype>ArticleDetails</requesttype> <requestlanguage>DE</requestlanguage> <requestformat>XML</requestformat> <articlemanufacturer>Targus Bags</articlemanufacturer> <articlereference>TCM005EU</articlereference> </request> </upgradeboxrequest></pre>

Function	The next example shows the SubUser Mode command for creating the "Public Testuser" for the common upgradebox account (this example can not be used directly in the testing enviroment due to the missing password, but you can work with the SubUser Mode functions with your own SubUSer 0000 password!)
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0000</subid> <pass>[PASSSSWORD]</pass> </authorization> <customvariables /> <request> <requesttype>SubUserAdd</requesttype> <requestlanguage>EN</requestlanguage> <requestformat>XML</requestformat> <subusername>Public Testuser</subusername> <subuserpassword>testpassword</subuserpassword> <subuserremarks>Testuser for usage with the upgbAPI documentation examples</subuserremarks> </request> </upgradeboxrequest></pre>

Function	Temporarily deactivate a subuser
XML Request	<pre><?xml version="1.0" encoding="utf-8" ?> <upgradeboxrequest> <authorization> <userid>12873</userid> <subid>0000</subid> <pass>[PASSSSWORD]</pass> </authorization> <customvariables /> <request> <requesttype>SubUserAdd</requesttype> <requestlanguage>EN</requestlanguage> <requestformat>XML</requestformat> <subuserid>001</subuserid> <subuserstatus>inactive</subuserstatus> </request> </upgradeboxrequest></pre>

6. Version history

Changes Version 1.3:

- Added Origin Storage Model Memory Configuration Details

Changes Version 1.2.1:

- Corrected link to the test-version (change .eu to .net)

Changes Version 1.2:

- Added function "ModelTypesList"
- Added fields to ModelInfo: haspicture, hasdatasheet, datasheetlanguage, has_manual, manuallanguage
- Added function "ModelPicture"
- Added function "ModelDatasheet"
- Added function "ModelManual"
- Added parameter requirehasmodeltype to the function "ModelManufacturer"
- Added parameter restrictmodeltype to the function "FreeTextModels"