Pubhub Web Services

_____

Supplier API Integration Guide

Suppliers / Publishers

January 2013

v1.2

TABLE OF CONTENTS

Date	Version	Comment
2011-07-01	1.0	First version of service released
2012-05-09	1.1	QA disabled at the moment
2013-01-22	1.2	New endpoint returning individual transactions added. Minor modifications to existing end point (SupplierServices.svc/salesreport)

DEFINITION OF TERMS

PubHub: Publizon Distribution Hub

License Key: A GUID that identifies you as a supplier. Please contact Publizon to get a license key.

HOW TO ACCESS THE WEB SERVICE

Pubhub exists in two separate environments, each with their own server, code and database. Please use the web service located at service.qa.pubhub.dk when developing and testing!

In order to have relevant data to test against a copy of the production database will regularly be transferred to the test environment. Because of this you can use the same license key in both environments (unless the supplier has been created in production after the copy has been transferred to test).

Please contact Publizon to get a license key.

Test / development environment domain:

http://service.qa.pubhub.dk

Production environment domain:
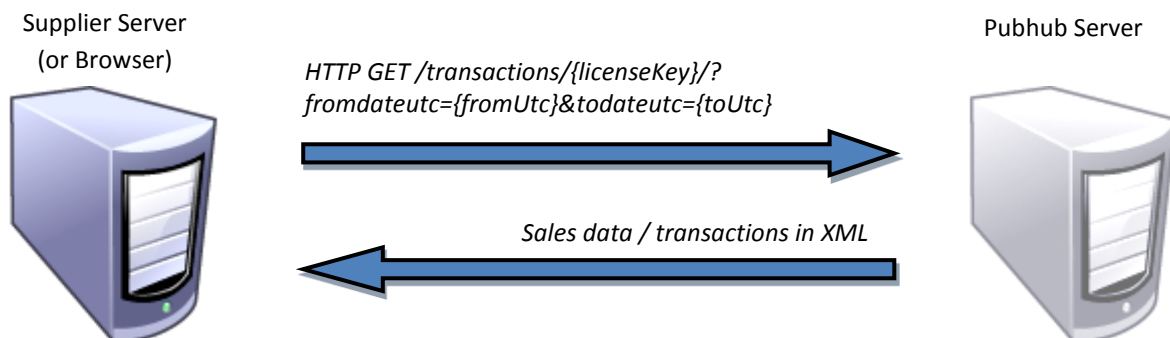
https://service.pubhub.dk

Attention: Remember to use SSL/https in production.

TRANSACTION LIST

The following end point returns individual transactions for the supplier / publisher with the specified license key in the specified period:

[Test/Prod Domain]/SupplierServices.svc/transactions/{licenseKey}/?fromdateutc={fromUtc}&todateutc={toUtc}

You can setup a server to automatically retrieve, process and transfer sales data into your own system or you can just see the sales data directly in a browser:

Supplier Server
(or Browser)

Pubhub Server

HTTP GET /transactions/{licenseKey}/?
fromdateutc={fromUtc}&todateutc={toUtc}

Sales data / transactions in XML

licenseKey: Your (the supplier's) private identification key – a GUID, e.g. 7871DA94-EFA6-4152-9A94-22F9D9AF52B8. It's optional to enclose the GUID with {}. The license key is issued by Publizon.

fromdateutc: The UTC date and time to retrieve transactions <u>from</u> – e.g. use "2012-12-31 23:00:00" to get transactions from (and including) the 1st of january 2013 (danish time).

todateutc: The UTC date and time to retrieve transactions <u>to</u> – e.g. use "2013-01-31 23:00:00" to get transactions to (and <u>not</u> including) the 1st of february 2013 (danish time).

(The UTC format has been chosen to avoid future complications with different time zones)

Below is an example of a complete (url-encoded) URL:

https://service.pubhub.dk/SupplierServices.svc/transactions/7871DA94-EFA6-4152-9A94-22F9D9AF52B8/?fromdateutc=2012-12-31%2022:00:00&todateutc=2013-01-31%2022:00:00

This URL doesn't work (will produce a http status code 403) because the license key is not valid!

Below is an example of the XML output data that the end point will produce:

```xml
<SalesReport xmlns="http://service.pubhub.dk" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <Transactions>
        <Transaction>
            <Identifier>0123456789012</Identifier>
            <Title>Imaginary</Title>
            <Authors>Anna Grue</Authors>
            <BookType>Ebog</BookType>
            <Imprint>Book World</Imprint>
            <PublicationDate>2010-04-09T00:00:00</PublicationDate>
            <CreatedDateUTC>2010-05-07T00:00:00</CreatedDateUTC>
            <LastUpdatedUTC>2011-06-08T10:45:33.36</LastUpdatedUTC>
            <SupplierPriceBeforeDiscount>82,00</SupplierPriceBeforeDiscount>
            <SupplierPriceAfterDiscount>82,00</SupplierPriceAfterDiscount>
            <RetailerPriceBeforeDiscount>86,10</RetailerPriceBeforeDiscount>
            <RetailerPriceAfterDiscount>86,10</RetailerPriceAfterDiscount>
            <RecommendedRetailerPrice>0,00</RecommendedRetailerPrice>
            <TransactionFee>4,10</TransactionFee>
            <CustomerId>123</CustomerId>
            <CustomerName>Nykøbing Bibliotek</ CustomerName >
            <IsLoan>true</IsLoan>
        </Transaction>
        <Transaction>
            <Identifier>9876543210210</Identifier>
            <Title>Best of John Doe</Title>
            <Authors>John Doe</Authors>
            <BookType>Ebog</BookType>
            <Imprint>Book World</Imprint>
            <PublicationDate>2011-01-07T00:00:00</PublicationDate>
            <CreatedDateUTC>2010-05-07T00:00:00</CreatedDateUTC>
            <LastUpdatedUTC>2011-01-03T13:21:06.36</LastUpdatedUTC>
            <SupplierPriceBeforeDiscount>81,19</SupplierPriceBeforeDiscount>
            <SupplierPriceAfterDiscount>81,19</SupplierPriceAfterDiscount>
            <RetailerPriceBeforeDiscount>85,25</RetailerPriceBeforeDiscount>
            <RetailerPriceAfterDiscount>85,25</RetailerPriceAfterDiscount>
            <RecommendedRetailerPrice>0,00</RecommendedRetailerPrice>
            <TransactionFee>4,06</TransactionFee>
            <CustomerId>321</CustomerId>
            <CustomerName>Ebogs Forhandler ApS</ CustomerName >
            <IsLoan>false</IsLoan>
        </ Transactions>
    </ Transactions>
</SalesReport>
```

A Transaction element represents one transaction (debit or credit). The prices will be positive in case of a sale/loan and negative in case of a cancellation.

Field/Element	Data type (max length)	Description
Identifier*	String(50)	Identifier (ISBN10, ISBN13 or GTIN13)
Title*	String(150)	Title of the e-book or audiobook
Athors*	String(1000)	Comma separated list of authors
BookType*	String(10)	"Ebog" or "Lydbog" (E-book or Audiobook)
Imprint	String(100)	The name of an imprint or brand under which the book is issued
PublicationDate	DateTime(19)	Date of publication in the format YYYY-MM-DDT00:00:00. The time part will in this case always be 00:00:00
CreatedDateUTC*	DateTime(19)	UTC date and time (specified in ISO 8601) of when the book was created in Pubhub.
LastUpdatedUTC*	DateTime(19)	UTC date and time (specified in ISO 8601) of when the book was last updated in Pubhub
SupplierPriceBeforeDiscount*	double	The supplier's unit price before an eventual discount. The price: - is the unit amount the supplier will receive from Publizon if no discount has been given - is equal to RetailerPriceBeforeDiscount subtracted by TransactionFee. - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 82,00
SupplierPriceAfterDiscount*	Double	The supplier's unit price after an eventual discount. The price: - is net unit amount the supplier will receive from Publizon - is equal to RetailerPriceAfterDiscount subtracted by TransactionFee - will be equal to SupplierPriceBeforeDiscount if no discount has been given - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 82,00
RetailerPriceBeforeDiscount*	Double	The retailer's unit price before an eventual discount. The price: - is the unit amount Publizon will receive from the retailer if no discount has been given. - is equal to SupplierPriceBeforeDiscount + TransactionFee - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 86,10
RetailerPriceAfterDiscount*	Double	The retailer's unit price after an eventual discount. The price: - is the net unit amount Publizon will receive from the retailer - is equal to SupplierPriceAfterDiscount + TransactionFee - will be equal to RetailerPriceBeforeDiscount if no discount has been given - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 86,10
RecommendedRetailPrice	Double	The supplier's suggested/recommended retail price (also called list price). The price: - is per unit - is in DKK - contains 2 digits to the right of the decimal point (,)

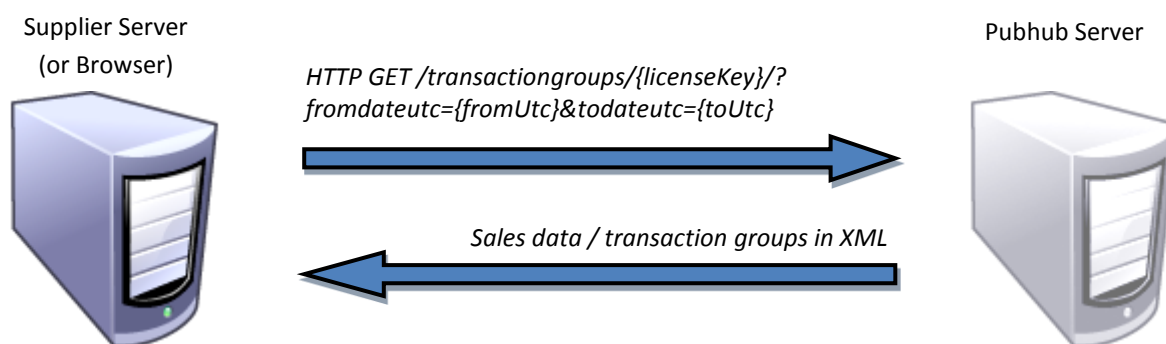| TransactionFee*	Double	The unit fee that Publizon charges per sold unit. The fee: - is per unit - is in DKK - contains 2 digits to the right of the decimal point (,)
CustomerId	Int	Id of the customer (retailer or library)
CustomerName	String(50)	Name of the customer (retailer or library)
IsLoan	Boolean	"true" if loan / library order and "false" if not loan / library order

TRANSACTION GROUP LIST

The following end point returns transaction groups for the supplier / publisher with the specified license key in the specified period:

[Test/Prod Domain]/SupplierServices.svc/transactiongroups/{licenseKey}/?fromdateutc={fromUtc}&todateutc={toUtc}

A group of transactions represents one or more transactions on the same title / isbn with exact same prices (supplier price before and after discount, retailer pris before and after discount and transaction fee).

You can setup a server to automatically retrieve, process and transfer sales data into your own system or you can just see the sales data directly in a browser:

Supplier Server (or Browser) — *HTTP GET /transactiongroups/{licenseKey}/? fromdateutc={fromUtc}&todateutc={toUtc}* → **Pubhub Server**

← *Sales data / transaction groups in XML*

INPUT

licenseKey: Your (the supplier's) private identification key – a GUID, e.g. 7871DA94-EFA6-4152-9A94-22F9D9AF52B8. It's optional to enclose the GUID with {}. The license key is issued by Publizon.

fromdateutc: The UTC date and time to retrieve transaction groups <u>from</u> – e.g. use "2012-12-31 23:00:00" to get transaction groups from (and including) the 1st of january 2013 (danish time).

todateutc: The UTC date and time to retrieve transaction groups <u>to</u> – e.g. use "2013-01-31 23:00:00" to get transaction groups to (and <u>not</u> including) the 1st of february 2013 (danish time).

(The UTC format has been chosen to avoid future complications with different time zones)

Below is an example of a complete (url-encoded) URL:

https://service.pubhub.dk/SupplierServices.svc/transactiongroups/7871DA94-EFA6-4152-9A94-22F9D9AF52B8/?fromdateutc=2012-12-31%2023:00:00&todateutc=2013-01-31%2023:00:00

This URL doesn't work (will produce a http status code 403) because the license key is not valid!

OUTPUT

Below is an example of the XML output data that the end point will produce:

```xml
<SalesReport xmlns="http://service.pubhub.dk" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <TransactionGroups>
        <TransactionGroup>
            <Identifier>0123456789012</Identifier>
            <Title>Imaginary</Title>
            <Authors>Anna Grue</Authors>
            <BookType>Ebog</BookType>
            <Imprint>Book World</Imprint>
            <PublicationDate>2010-04-09T00:00:00</PublicationDate>
            <CreatedDateUTC>2010-05-07T00:00:00</CreatedDateUTC>
            <LastUpdatedUTC>2011-06-08T10:45:33.36</LastUpdatedUTC>
            <NumberOfTransactions>3</ NumberOfTransactions>
            <SupplierPriceBeforeDiscount>82,00</SupplierPriceBeforeDiscount>
            <SupplierPriceAfterDiscount>82,00</SupplierPriceAfterDiscount>
            <RetailerPriceBeforeDiscount>86,10</RetailerPriceBeforeDiscount>
            <RetailerPriceAfterDiscount>86,10</RetailerPriceAfterDiscount>
            <RecommendedRetailerPrice>0,00</RecommendedRetailerPrice>
            <TransactionFee>4,10</TransactionFee>
            <TransactionFeeTotal>12,30</TransactionFeeTotal>
            <SupplierPriceAfterDiscountTotal>246,00</SupplierPriceAfterDiscountTotal>
            <RetailerPriceAfterDiscountTotal>258,30</RetailerPriceAfterDiscountTotal>
        </TransactionGroup>
        <TransactionGroup>
            <Identifier>9876543210210</Identifier>
            <Title>Best of John Doe</Title>
            <Authors>John Doe</Authors>
            <BookType>Ebog</BookType>
            <Imprint>Book World</Imprint>
            <PublicationDate>2011-01-07T00:00:00</PublicationDate>
            <CreatedDateUTC>2010-05-07T00:00:00</CreatedDateUTC>
            <LastUpdatedUTC>2011-01-03T13:21:06.36</LastUpdatedUTC>
            <NumberOfTransactions>1</NumberOfTransactions>
            <SupplierPriceBeforeDiscount>-81,19</SupplierPriceBeforeDiscount>
            <SupplierPriceAfterDiscount>-81,19</SupplierPriceAfterDiscount>
            <RetailerPriceBeforeDiscount>-85,25</RetailerPriceBeforeDiscount>
            <RetailerPriceAfterDiscount>-85,25</RetailerPriceAfterDiscount>
            <RecommendedRetailerPrice>0,00</RecommendedRetailerPrice>
            <TransactionFee>-4,06</TransactionFee>
            <TransactionFeeTotal>-4,06</TransactionFeeTotal>
            <SupplierPriceAfterDiscountTotal>-81,19</SupplierPriceAfterDiscountTotal>
            <RetailerPriceAfterDiscountTotal>-85,25</RetailerPriceAfterDiscountTotal>
        </TransactionGroup>
    </TransactionGroups>
</SalesReport>
```

DESCRIPTION OF FIELDS

Field/Element	Data type (max length)	Description
Identifier*	String(50)	Identifier (ISBN10, ISBN13 or GTIN13)
Title*	String(150)	Title of the e-book or audiobook
Athors*	String(1000)	Comma separated list of authors
BookType*	String(10)	"Ebog" or "Lydbog" (E-book or Audiobook)
Imprint	String(100)	The name of an imprint or brand under which the book is issued
PublicationDate	DateTime(19)	Date of publication in the format YYYY-MM-DDT00:00:00. The time part will in this case always be 00:00:00
CreatedDateUTC*	DateTime(19)	UTC date and time (specified in ISO 8601) of when the book was created in Pubhub.
LastUpdatedUTC*	DateTime(19)	UTC date and time (specified in ISO 8601) of when the book was last updated in Pubhub
NumberOfTransaction*	Int	The net amount sold or cancelled of the given book in the requested period at the given prices.
SupplierPriceBeforeDiscount*	double	The supplier's unit price before an eventual discount. The price: - is the unit amount the supplier will receive from Publizon if no discount has been given - is equal to RetailerPriceBeforeDiscount subtracted by TransactionFee. - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 82,00
SupplierPriceAfterDiscount*	Double	The supplier's unit price after an eventual discount. The price: - is net unit amount the supplier will receive from Publizon - is equal to RetailerPriceAfterDiscount subtracted by TransactionFee - will be equal to SupplierPriceBeforeDiscount if no discount has been given - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 82,00
RetailerPriceBeforeDiscount*	Double	The retailer's unit price before an eventual discount. The price: - is the unit amount Publizon will receive from the retailer if no discount has been given. - is equal to SupplierPriceBeforeDiscount + TransactionFee - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 86,10
RetailerPriceAfterDiscount*	Double	The retailer's unit price after an eventual discount. The price: - is the net unit amount Publizon will receive from the retailer - is equal to SupplierPriceAfterDiscount + TransactionFee - will be equal to RetailerPriceBeforeDiscount if no discount has been given - is in DKK - contains 2 digits to the right of the decimal point (,) Example: 86,10
RecommendedRetailPrice	Double	The supplier's suggested/recommended retail price (also called list price). The price: - is per unit

		- is in DKK - contains 2 digits to the right of the decimal point (,)
TransactionFee*	Double	The unit fee that Publizon charges per sold unit. The fee: - is per unit - is in DKK - contains 2 digits to the right of the decimal point (,)
TransactionFeeTotal*	Double	The total transaction fee that Publizon charges for the net amount sold of the given book in the requested period at the given prices. The total transaction fee: - is equal to NumberOfTransactions multiplied by TransactionFee - is in DKK - contains 2 digits to the right of the decimal point (,)
SupplierPriceAfterDiscountTotal*	Double	The total supplier price after eventual discounts. The price: - is the total amount the supplier will receive from Publizon - is equal to NumberOfTransactions multiplied by SupplierPriceAfterDiscount - is in DKK - contains 2 digits to the right of the decimal point (,)
RetailerPriceAfterDiscountTotal*	Double	The total retailer price after eventual discounts. The price: - is the total net amount the Publizon will receive from the retailer - is equal to NumberOfTransactions multiplied by RetailerPriceAfterDiscount - is in DKK - contains 2 digits to the right of the decimal point (,)