Guidelines for Implementation of a SOAP-Client for the Webservice API of Speed4Trade CONNECT

Version 1.94

This document was automatically translated using Google Translation. Please excuse any non-human expressions und funny results. The translation should be sufficient to get the required information transferred. All images are still in German language.

Contents

Contents **2**
1 Introduction **6**
2 Speed4Trade CONNECT **7**
2.1 Speed4Trade CONNECT as a data hub for online trading 7
2.2 Speed4Trade CONNECT as middleware 8
2.2.1 System configuration of a "complete system" (ERP + Speed4Trade CONNECT) 8
3 The software development project SOAP client **10**
3.1 Communication Protocol 10
3.1.1 Integration Request XML in SOAP envelope 12
3.2 Security 13
3.2.1 Encrypted communication 13
3.2.2 Server Authentication 14
3.2.3 Client Authentication 14
3.3 Development Framework 14
3.4 Program Logic 15
3.5 Requirements Analysis 16
4 Functional overview **17**
4.1 Basic settings, master data and offer Listing 17
4.2 Quotation and Order Workflow 18
5 Important basic settings - preparatory API Calls **19**
5.1 Clients 21
5.1.1 Webservice Call: fetchMandators (clients) 21
5.1.1.1 Languages 21
5.1.1.2 Currencies 21
5.2 Sales channels 22
5.3 SubChannels 22
5.3.1 Webservice Call: fetchSubChannels (Sub - Channels) 22
5.4 Accounts 23
5.4.1 Webservice Call: fetchAccounts (accounts) 23
5.5 Text block types 24
5.5.1 Webservice Call: fetchTextModuleTypes (text block types) 25
5.6 Image block types 26
5.6.1 Webservice Call: fetchImageModuleTypes (image block types) 27
5.7 Media Management 27
5.8 Application of image, text and media modules in offers 28
5.9 Centralized Attribute Management (CAM) 29
5.9.1 Webservice-Call: fetch / updateAttributes (central attributes and values) 30
5.9.2 Webservice-Call: fetch / updateAttributeSets (central attribute groupings) 30
5.10 Pricelists (updatePricelist) 31
5.11 Return Categories 33
5.11.1 Webservice Call: fetchDeliveryChargeCategories / updateDeliveryChargeCategories (Shipping cost categories) 33
5.11.1.1 Returns 33
5.12 Delivery Times 34
5.12.1 Webservice Call: fetchDeliveryTimes / updateDeliveryTimes (delivery times 34

5.13 Manufacturer ... 35
5.13.1 Webservice Call: fetchManufacturers / updateManufacturers (manufacturer ... 35
5.14 Warehouse / Stock ... 36
5.14.1 Webservice Call: fetchWarehouses (stock) ... 36
6 Product Groups ... 36
6.1 Webservice Call: UpdateCategories / fetchCategories (departments) ... 37
7 Item master data ... 38
7.1 Item Basic data ... 38
7.1.1 Basic data - general ... 39
7.1.2 Basic Data - Text Management ... 40
7.1.3 Basic Data - Image Management ... 41
7.2 Webservice Call: update items (item master data) ... 42
7.2.1 Article -> Basic Data -> General ... 42
7.2.2 Item -> Basic Data -> Additional Articles ... 44
7.2.3 Item -> Basic Data -> article-bundle ... 44
7.2.4 Article variants ... 44
7.2.5 Item Attributes (Centralized Attribute Management - CAM) ... 44
7.3 Webservice-Call: fetchItems ... 45
8 Sales channel - specific data ... 46
8.1 Item - Description and Price ... 46
8.2 Amazon marketplace ... 47
8.2.1 Webservice Call: updateItems (Amazon -specific data) ... 47
8.2.1.1 Item -> Amazon -> general ... 48
8.2.1.2 Item -> Amazon -> Images ... 48
8.2.1.3 Item -> Amazon -> product data ... 49
8.2.1.4 Item -> Amazon -> Category Data ... 49
8.3 Marketplace eBay ... 51
8.3.1 Webservice Call: updateItems (eBay-specific data) ... 52
9 Overview of marketplace-specific characteristics ... 54
10 Inventory Management ... 55
10.1 stock transfer by item master data ... 56
10.1.1 Webservice Call : update items (itemset) ... 57
10.1.2 Item -> Basic Data -> inventory management ... 57
10.2 stock transfer by separate stock update ... 58
10.2.1 Webservice-Calls: updateStockCount (Artikelbestände) ... 58
11 Order Management ... 59
11.1 Order Workflow without an external system ... 59
11.2 Order Workflow with an external system (ERP / MRP) ... 60
11.2.1 Default Job Workflow ... 61
11.2.1.1 Duplicate Address Check ... 62
11.2.1.2 orders with customer comments ... 63
11.2.2 Order workflow for eBay orders without a completed eBay Checkout ... 63
11.2.3 Order workflow for Amazon orders - waiting for e- mail ... 64
11.3 Webservice-Call: fetchOrders ... 65
11.3.1 Relevant fields in order ... 65
11.3.1.1 Orders -> Basic Data ... 65
11.3.1.2 Orders -> packages ... 66

11.3.1.3 Job Status : regrouped jobs on eBay 66
11.4 Webservice Call : update orders 68
12 Additional Functions 69
12.1 Initial Notification-Mail to buyers 69
12.2 Payment balance 69
12.2.1 Webservice-Call: fetchPayments 69
12.3 Delivery of orders via FBA 70
12.4 Partial deliveries 70
12.5 After Sales 70
12.5.1 eBay 70
12.5.2 Amazon 70
13 Remote control of Speed4Trade CONNECT 71
13.1 General 71
13.2 Workflow for automatically adjusting 71
13.3 Balance of new product data into marketplaces 72
13.4 Show masks of Speed4Trade CONNECT in a container 72
13.4.1 Structure of the mask address for the automatic login 72
13.4.2 Information about authentication 72
13.5 Hiding the framework of Speed4Trade CONNECT WEB Templates 73
13.6 Definition of a callback address 73
13.6.1 Parameters callback not set 73
13.6.2 Parameters callback = no 74
13.6.3 Parameters callback put on a different mask 74
14 Summary 75
14.1 Workflow overview 76

List of Figures

Illustration 1: Scheme of Speed4Trade CONNECT as a data hub for online trading 7
Illustration 2: System design of a "complete system" 8
Illustration 3: HTTPS communication of XML data using the SOAP protocol 11
Illustration 4: Overview of Functions - Basic Settings, Master Data and Listing 17
Illustration 5: Feature List - quote and order workflow 18
Illustration 6: Mandatory fields are highlighted with a "yellow" bar 19
Illustration 7: Translation option for the Speed4Trade CONNECT Online Manual 20
Illustration 8: Client - management 21
Illustration 9: Accountverwaltung 23
Illustration 10: text block types 24
Illustration 11: faceplate types 26
Illustration 12: Application of image, text and media modules in offers 28
Illustration 13: Item Attributes 29
Illustration 14: Integration samples for Attributes 30
Illustration 15: Pricelists 31
Illustration 16: Usage scenario of Basic Price Lists (BPL) 32
Illustration 17: return categories 33
Illustration 18: Delivery Times 34
Illustration 19: Manufacturer 35
Illustration 20: Stock 36
Illustration 21: classes of goods 37
Illustration 22: item basic data 39
Illustration 23: item text management 40
Illustration 24: item Images Management 41
Illustration 25: Item data - Amazon 47
Illustration 26: eBay Category 51
Illustration 27: Stock management 55
Illustration 28: Navigation bar 59
Illustration 29: offer workflow 59
Illustration 30: order workflow 59
Illustration 31: order data 60
Illustration 32: order workflow 62
Illustration 33: Workflow for automatically adjusting 71
Illustration 34: Overview diagram of the workflow 76

1 Introduction

The present guidelines for implementing a SOAP client for the Webservice API of Speed4Trade CONNECT gives you a comprehensive overview of the implementation of an own connection to the Webservice of the Speed4Trade GmbH.

The aim of this document is to:

- you a general overview of the operation of the release.
- to sensitize you that the implementation of a SOAP client is a full software development project.
- show you the basic workflow of Speed4Trade CONNECT.
- The need for mapping the workflows Webservice calls to define.
- The connection to refine step by step in the integration depth.

The guide can be read as follows. First you get per chapter:

- a general explanation of the function using the Speed4Trade CONNECT clients,
- then the Webservice calls are declared to be a minimum standard connection
- and, finally given evidence of extended connection with additional options.

You should first start with the implementation of a standard connection. This has the advantage that you will need no deep knowledge on the functioning of connected marketplaces (eBay, Amazon, etc.) or online stores. The default connection is limited to the preliminary reading of basic and master data, the passing of Article basic data, inventory changes, as well as the collection of orders and setting new order status. By doing Weis, you can achieve rapid project successes and get train to train more certainty in the application of web service calls. In a second step, more complex calls and special functions can be realized.

Now read what the Speed4Trade CONNECT which functions and features has the software and as an example of a "complete system" with ERP / ERP, Speed4Trade CONNECT is designed as middleware and online sales channels.

2 Speed4Trade CONNECT

The Speed4Trade CONNECT is a client-server based application software for companies engaged in Internet commerce. It allows to sell products and goods of all kinds of highly automated via a variety of only sales channel like your online store (multi-store capability), eBay, Amazon, Yatego, Rakuten, Shopgate etc. The software combines the sales process through the various online channels centrally in one system and thus allows the development of additional distribution channels and efficient multi-channel distribution on the Internet. The software controls all aspects of professional online trading: Articles, preparation for the product presentation on the online platforms, listing in the various user accounts, buyer communication, order processing, to supporting the shipping process and after-sales processes.

2.1 Speed4Trade CONNECT as a data hub for online trading

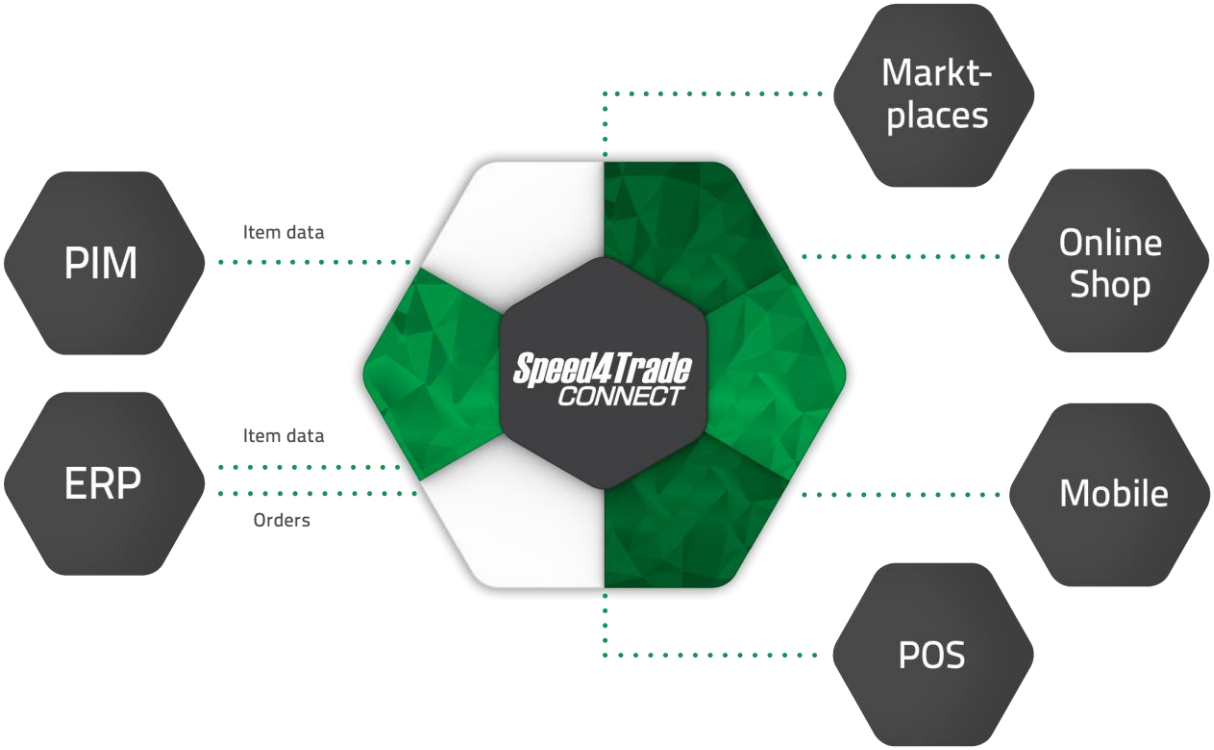

Illustration 1: Scheme of Speed4Trade CONNECT as a data hub for online trading

Among the features of the Speed4Trade CONNECT include, e.g. an ergonomic, user-friendly interface, a real Client-/Server-Technology and performant web service connections. The user interface is geared towards common standards established business applications, making it very intuitive to use. The software is scalable and provides important components such as users, clients, countries, languages, currencies in "variety" from, making the system has been designed from the outset for growth and the expansion of the online business. A special feature is the interactive inventory balance (IABA). This automatically and sales channel across resembles the inventories of items in the online deals from

according to the availability of goods . Thus Fills offer stocks back on and therefore controls the smooth multi -channel sales. Numerous automated mechanisms such as the timed setting , order pick , buyer notifications or payment reconciliations replace the manual intervention in the business processes , allowing the software offers immense potential cost savings while improving the process quality.

The system can be operated as an in-house solution in a corporate network or as a hosted solution from different locations . In addition, the solution is client - customizable and can therefore be tailored to the different requirements and business processes (Customizing)

2.2 Speed4Trade CONNECT as middleware

The Speed4Trade CONNECT solution - in addition to its use as a stand-alone version - are used as so-called middleware between an existing ERP / ERP system and the various "online worlds". The connection of external systems (eg ERP / ERP) via the Speed4Trade-CONNECT-API-Webservice interface. Thus can be automated exchanged between the systems data such as articles, stocks and orders, which forms the basis for integrated sales processes and low process costs.

2.2.1 System configuration of a "complete system" (ERP + Speed4Trade CONNECT)

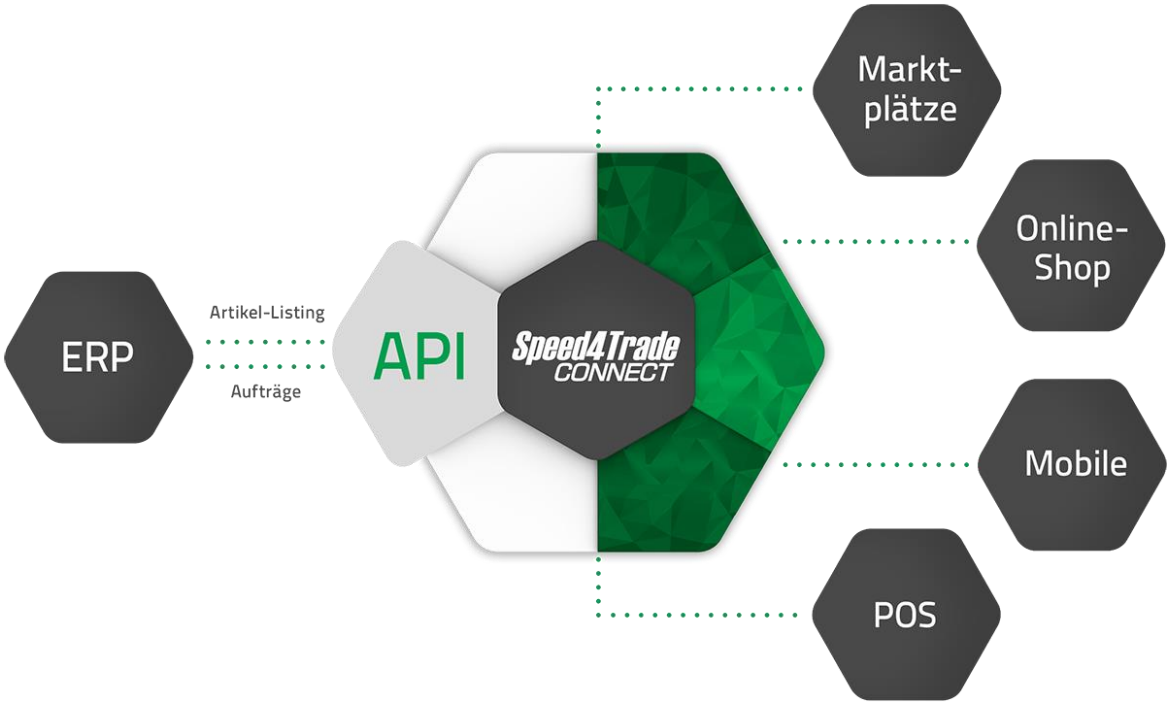

Illustration 2: System design of a "complete system"

When implementing roughly following processes are to be implemented :

- Basic data such as images, description, price lists of an item , etc. passed to the Speed4Trade CONNECT from the external system.
- In the Speed4Trade CONNECT in the simple integration model , the further sales channel - specific processing of the data , for example, assignment of categories and attributes.
- The external system regularly provides current stock values and, if the price changes to the articles.
- After the final preparation of the article in the Speed4Trade CONNECT these are set in the sales channels as " current offers".
- New orders will be picked up by the Speed4Trade CONNECT from the sales channels.
- The external system retrieves these orders out in a defined status and in turn, updates the job status in the Speed4Trade CONNECT after certain events (receipt of payment , shipping of goods).
- The Speed4Trade CONNECT synchronizes the job status in the respective sales channels and informs the customer about the current status.

The list makes it clear that it is in developing a SOAP client by more than just a couple of scripts for data import and export. A SOAP client is a full program with its own program logic and must therefore be implemented with professional development tools and methods.

3 The software development project SOAP client

To use the Speed4Trade-CONNECT-Webservice, developers should have a good understanding of and experience with SOAP (Simple Object Access Protocol, see http://de.wikipedia.org/wiki/SOAP) bring. It is only an Internet connection necessary on the Webservice via TCP / IP can be addressed. In addition to the connection management with the Speed4Trade-CONNECT-Webservice, SOAP and XML tools should be available to create call - documents from the application logic out and read the information from the response document ('parse'). In addition, the URLs of the web service invocation points (endpoints), as well as 'User Name' and ' Password' and activation required.

A WSDL file (Web Services Description Language) with the relevant web service definitions is provided under the URL https://YOUR-SPEED4TRADE-CONNECT-SERVER/services/ecs-ws?wsdl
This is needed to develop applications for this Webservice. You specify the required structure of the SOAP messages, the function signatures, API Service Bindings 'and, service endpoints'.

With these descriptions, developers can develop a SOAP client to generate business object interfaces, and network stubs from it and make the work significantly. The request is finally formulated as a SOAP method call and sent to the SOAP service endpoint.

It is recommended to use the most current bindings (ecs-wsSoap12Binding) and Endpoints (ecs-wsHttpsSoap12Endpoint) for SOAP 1.2. The SOAP 1.1 support is obsolete and only still available for downward compability reasons.

3.1 Communication Protocol

For communication with the Webservice a combination of request and response data in XML format and the standard SOAP protocol is used. This by itself provides no way to versioning requests and responses, however, can easily be addressed on many systems.

For this reason, XML is chosen as a flexible, data format, and SOAP used as easy to speaking container protocol to transport requests and responses from the web service.

Please make ensure that a recent and secure SSL version is used. Once a security protocol or process version becomes uncertain, it will not be supported any longer (Cipher method). There will be no notification.

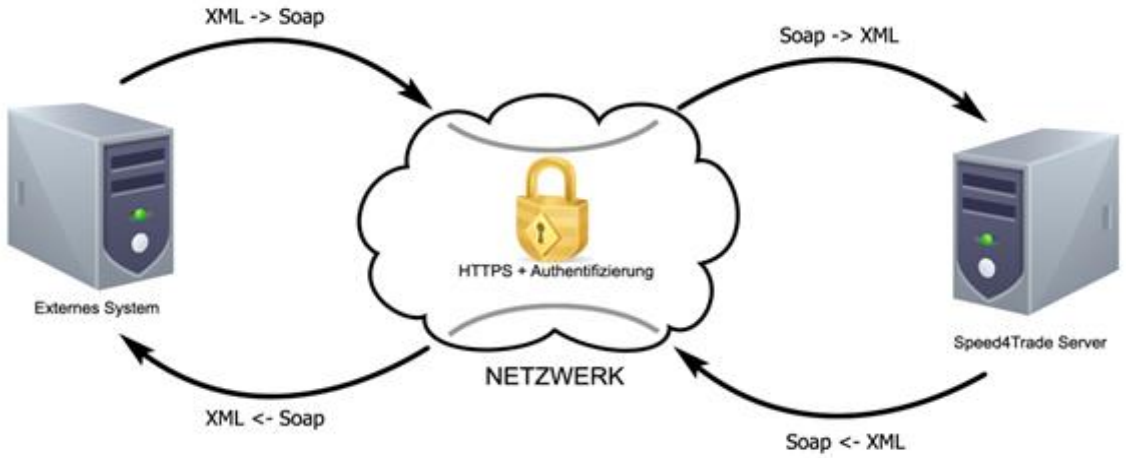

Illustration 3: HTTPS communication of XML data using the SOAP protocol

An exemplary structure of the XML data for the communication is specified as follows:

Example Request: reading the stored in Speed4Trade CONNECT system accounts

```
<request method="fetchAccounts" version="1.0.0">
   <filter>
      <mandator_id filter_method="=" filter_value="1" />
      <last_changed from_date="2000-01-01T00:00:00.000+01:00"
     to_date="2012-12-31T00:00:00.000+01:00" />
      <created_at from_date="2000-01-01T00:00:00.000+01:00"
     to_date="2012-12-31T00:00:00.000+01:00" />
   </filter>
</request>
```

Example Response: Assigned Accounts

```
<response method="fetchAccounts" version="1.0.0">
   <report return_code="0"/>
   <account>
      <account_id>25217</account_id>
      <mandator_id>1</mandator_id>
      <account_identifier>eBay-Account</account_identifier>
      <channel>
         <channel_id>2000</channel_id>
         <channel_identifier>eBay</channel_identifier>
      </channel>
      <sub_channel>
         <sub_channel_id>2100</sub_channel_id>
         <sub_channel_identifier>eBay.de</sub_channel_identifier>
      </sub_channel>
      <email_account>
         <email_account_id>24594</email_account_id>
```

```
<email_account_identifier>Speed4Trade</email_account_identifier>
         <email_address>contact@speed4trade.com</email_address>
      </email_account>
      <active>true</active>
      <order_number_prefix>Bay</order_number_prefix>
   </account>
</response>
```

As data types only standardized, defined by the XML Schema data type definition types are used (see also the API documentation at: https://partner.speed4trade.com/entwickler/api-webservice/ -> Introduction -> Explanation of the field list).

Hint:
If the Webservice received input data that cannot be interpreted, an exception is triggered by the server that can be caught and **handled by the client**. Two cases must be distinguished. Firstly, if the SOAP request is structurally wrong and cannot be processed and on the other hand if the request is structurally correct content but wrong though. That is, if, for example, an order is requested with the wrong account.

3.1.1 Integration Request XML in SOAP envelope

As noted in Figure 3, the request data in XML format to be sent within the SOAP envelope. For this purpose the XML data to be inserted either in a CData block or as pure XML data, however, must in this case, a masking of the data is carried. The masking can happen by means of existing in many programming languages EncodeXml function. Subsequently, each example of SOAP envelope, request data in CData block and masked data.

Example SOAP-Envelope: simplified construct

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
   xmlns:web="http://webservice.web.speed4trade.com">
   <soap:Header/>
   <soap:Body>
      <web:fetchMandators>
         <web:requestData>
            <!-- Request-XML -->
         </web:requestData>
      </web:fetchMandators>
   </soap:Body>
</soap:Envelope>
```

In this envelope data can now be integrated as CData block. To this end, here is a small example on the basis of calls fetchMandators.

Example CData block: Read the client (I)

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
   xmlns:web="http://webservice.web.speed4trade.com">
   <soap:Header/>
   <soap:Body>
      <web:fetchMandators>
         <web:requestData>
            <![CDATA[<?xml version="1.0" encoding="UTF-8"?>
               <request method="fetchMandators" version="1.0.0" />
            ]]>
         </web:requestData>
      </web:fetchMandators>
   </soap:Body>
</soap:Envelope>
```

Example of masked Request: Reading the clients (II)

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
   xmlns:web="http://webservice.web.speed4trade.com">
   <soap:Header/>
   <soap:Body>
      <web:fetchMandators>
         <web:requestData>
            &lt;?xml   version=&quot;1.0&quot;   encoding=&quot;UTF-
8&quot;?&gt;
            &lt;request            method=&quot;fetchMandators&quot;
version=&quot;1.0.0&quot; /&gt;
         </web:requestData>
      </web:fetchMandators>
   </soap:Body>
</soap:Envelope>
```

3.2 Security

3.2.1 Encrypted communication

The use of the Webservice is only possible via a secured using TLS encryption HTTPS connection. This protection against eavesdropping and integrity of the transmitted data is guaranteed.

Furthermore, the communication thereby protected from various other attacks. One example is the replay attack, called in an attacker sends an intercepted message again to the recipient. This is detected by the use of sequence numbers and blocked such an attack. It should be mentioned here that the attacker can’t decrypt and understand so that packets retransmitted.

3.2.2 Server Authentication

To ensure the communicating with the correct server and not an unauthorized location, the server's certificate is used to authenticate in the correct way (see the documentation for the test environment at: https://partner.speed4trade.com/entwickler/tools/ssl-zertifikat-download/ -> Sichere Verbindung zur Speed4Trade-CONNECT-API).

Attention: The Test-Environment uses a certificate which is different from life installations.

3.2.3 Client Authentication

To ensure that only legitimate clients can communicate with the Webservice must register on this server via HTTP Basic authentication.

Although the access of this process are transmitted in the clear in the header of the message , it is well suited for use in this case, since it is easy to implement on the client side , and the transmitted access due to the communication via HTTPS are not already listened to can.

3.3 Development Framework

In particular, if an ERP own development environment to be used in advance of technical development following questions must be answered:

- is a full-fledged development environment:
  - Support of SOAP / XML messages
  - Automatic code generation for data objects from WSDL files
  - Versioning and Debugging for care and maintenance
- Data can be stored in a database (basic data, intermediate results, temporary query data, etc.)
- to log the web service calls via a log file is possible
- can for transmission errors , such as respond by resending the case of non availability
- a program logic can be built up over a number of API calls across stands next to the productive also a test system.

Remember that a Webservice API is intended for the automatic exchange of large amounts of data and must therefore work stable, performant and reliable. Reason, plan a modular construction of SOAP clients with standardized controller / adapter system. This also facilitates the update of the SOAP clients for extensions of the Webservice API, because these constantly evolving.

3.4 Program Logic

In addition to routines for data-only query, you should also have a minimum of program modules for the following tasks:

- code generator , which generates WSDL files from base classes in your programming language.
- module to authenticate the SOAP clients on Speed4Trade-CONNEC-Webservice.
- program for reading static data (e.g. client number). It is sufficient here usually the data updates, i.e. when needed or once e.g. to update weekly. These data should be cached in the ERP or SOAP client, so they do not need to be additionally loaded only when dynamic data exchange (e.g. when retrieving orders) via the Webservice.
- General routines for storing results or intermediate results in a database.
- routine for synchronizing data, which is needed for further queries.
- mapping routines to determine safe for matching of Speed4Trade CONNECT record IDs to the record numbers of its own solution, the correct assignment of the records.
- controller for processing results, events and errors with the option checked to work with and the API call , if necessary, to re-run this .
- Central logging routine to log all function calls, error and debug messages.

A sophisticated logging mechanism is even more important when the SOAP client running constantly in the background. Only by erroneous calls can be immediately detected and improved. It must not necessarily be completely rejected by a Webservice call. Just think, for example, the fact that you pass 20 articles and contains a faulty data. Without detailed logging, you have no chance of this error to determine a timely manner.

Note!
To simplify the interface development, there are optional XSD files available in the partner portal.

Attention! To ensure downward compatibility of your interface XSDs provided shall not be used for validation purposes.

For the development and failure analysis can Speed4Trade CONNECT side a logging of SOAP messages via the web service be turned on. Please contact to our support team. Attention! Turn on logging in live operation at the customer necessarily back out because the logging affects the speed of the overall system.

It should also be that not only transmitted via some web service calls data, but also other functions in the Speed4Trade CONNECT be triggered in sequence. If you set, for example, a shipment status in the job, it is immediately transmitted via a comparison with the market place automatically in the marketplace. Without high quality SOAP client and a professional logging is here on a permanent no stable solution to achieve.

In addition to the purely technical part, it is important to perform a technical analysis of needs well in advance of development. What kind of data and functions are needed at all and how Webservice calls, you can concentrate it?

3.5 Requirements Analysis

Before you begin developing a SOAP client, you should start, as with any project, with a very well planned requirements analysis. In the default configuration, we assume that the ERP-System/die ERP is the leading system. That is, the ERP actively transfers the product data (including images) and current stock levels at the Speed4Trade CONNECT, platform - specific data in Speed4Trade CONNECT to complement and set an offer to the sales channels. The ERP in turn actively collects the orders received from Speed4Trade CONNECT and is based on job run workflow status for delivered and paid for on. This is then passed through Speed4Trade CONNECT back to the sales channels (market places and shops).

The following questions will help you with the needs analysis continues:

- How many clients are available and are to be served?
- Which sales channels (eBay.de, Amazon.de, shops, etc.) are to be connected?
- What accounts there are per sales channel?
- What item master data are already available, which must be completed?
- Are there extended descriptions to the articles?
- Are the ERP system already platform - specific data (e.g. item attributes) available?
- Where and how are the pictures of the items stored (locally , accessing the URL)?
- If you are working with variants articles and these are already fully invested in ERP?
- Should be used with sets, cross-sell and accessories?
- What are the prices, special prices, EIA should be offered ; Basic rates are needed?
- How is the current inventory is determined, there are one or more bearings?
- What are the shipping and shipping cost categories structured?
- Are the products already sorted in groups of products and these can be used for assigning categories to your sales channels?
- What job workflows are defined (" Payment workflow " -> only pay delivery and / or " billing workflow " -> only provide pay) ?
- What are the job data and status changes can be processed and enforced?
- Which payment methods are used and where will the cash balance held?
- Where the transfer of order data to the shipping service will take place and how is the tracking ID supplied back to tracking?
- How are incoming customer data (duplicate check) assigned from the orders?

  By answering the above questions, you can determine your needs and specifically documented in use cases. Can then easily Based on the necessary API calls to be determined. Let's start with the "Preparatory API Calls" such as reading out the client ID, which is almost always required for each subsequent call.

4 Functional overview

4.1 Basic settings, master data and offer Listing

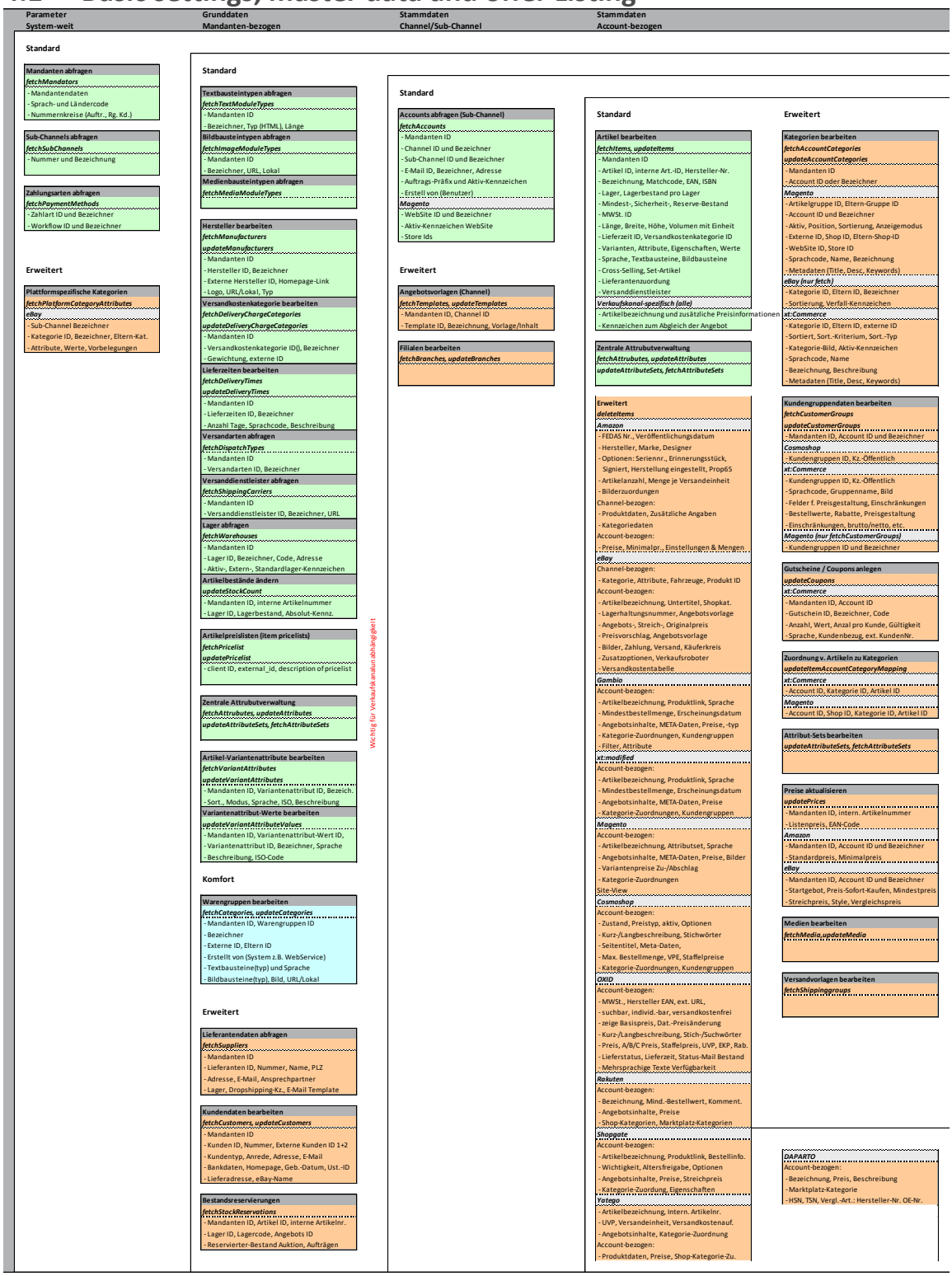

Illustration 4: Overview of Functions - Basic Settings, Master Data and Listing

4.2 Quotation and Order Workflow

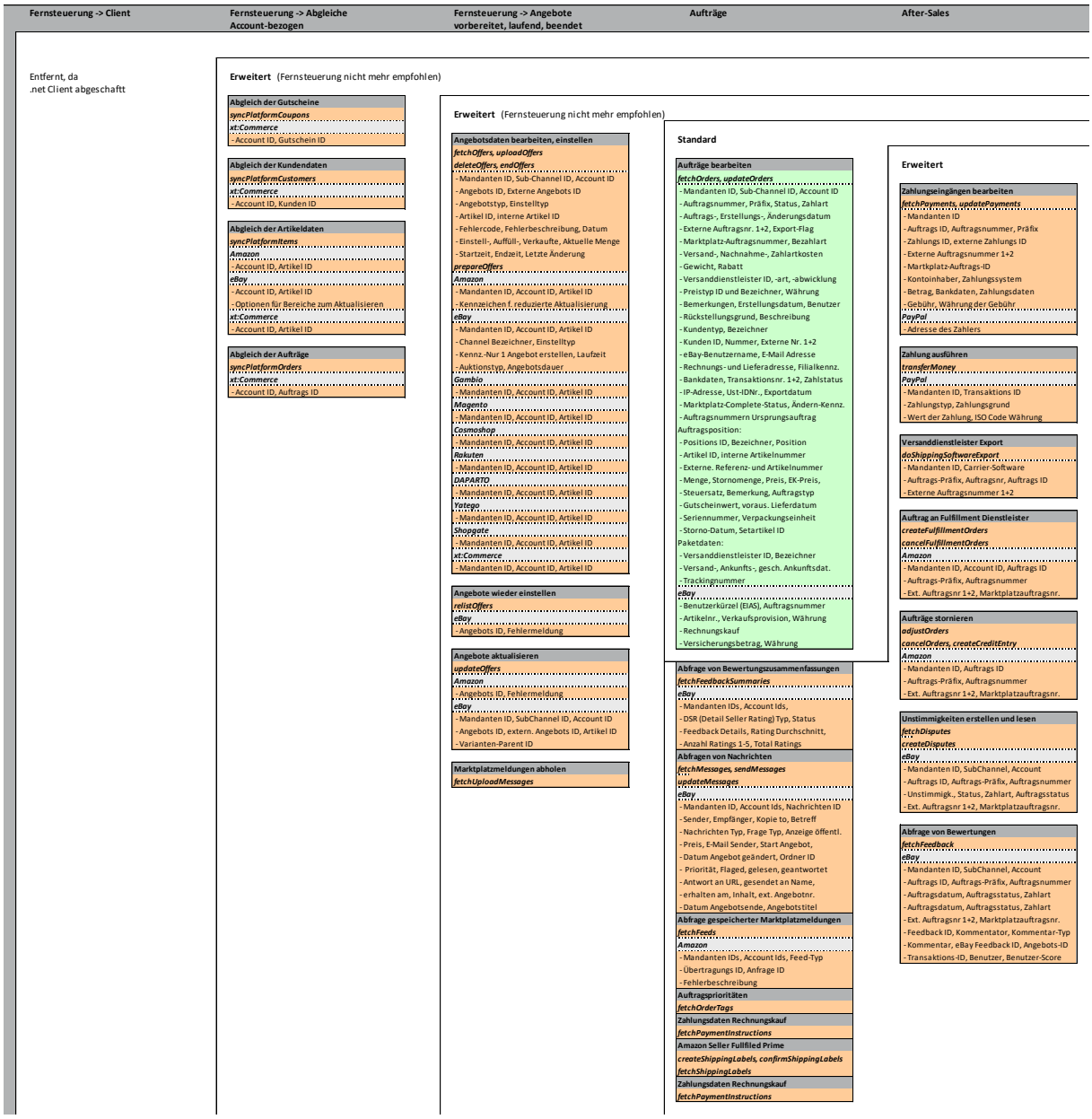

Illustration 5: Feature List - quote and order workflow

In the functional overviews the individual Webservice API calls are highlighted in each color. The color

- Green for the standard connection,
- blue for the convenience function of the product groups and
- red for the Advanced connection is.

You should first start with a default implementation. If necessary, add the comfort function on the allocation of products to product groups and only in a phase 2, integrated with advanced features.

The overview is available as an Excel file and can be downloaded at the following link:
https://partner.speed4trade.com/entwickler/integration-starten/integrationsleitfaden/

5 Important basic settings - preparatory API Calls

As has been already mentioned, there are some data, such as the client number that rarely or never change. This data should be determined independently from the current data exchange in advance and in the SOAP client or the ERP system e.g. are stored in a configuration table between. The detection and modification of this data is usually in Speed4Trade CONNECT. The update should be implemented so that, when needed, i.e. when changes or regularly for example, once a week. This increases the performance of the SOAP clients considerably when, for example previously not only to the clients also necessary number must be determined when updating an item.

Note!
Mandatory fields can be determined via the Speed4Trade CONNECT client. It is sufficient for the desired range, e.g. Articles, simply the "New" function call. In the entry screen appears, then all are required fields indicated with a "yellow" bar at the beginning of the input field.

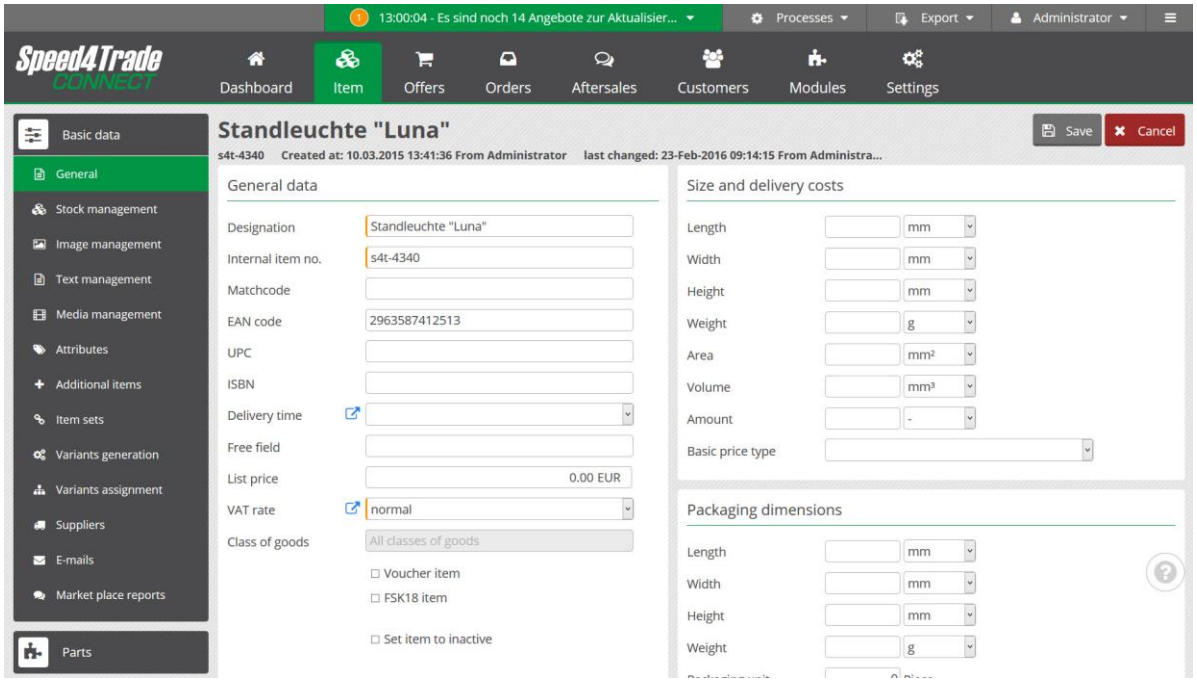

Illustration 6: Mandatory fields are highlighted with a "yellow" bar

While implementing the Webservice, it makes perfect sense to use the Speed4Trade-CONNECT-UI and the ***Speed4Trade CONNECT*** Online Manual to get a better understanding of the application and

the data fields used. To change the language of the ***Speed4Trade CONNECT Online Manual,*** navigate to the “Mehr”-Button on the to right and choose your preferred language which uses google translation service. Please be aware that this will lead to different translations in the manual compared to the English version of the Speed4Trade-CONNECT-UI.

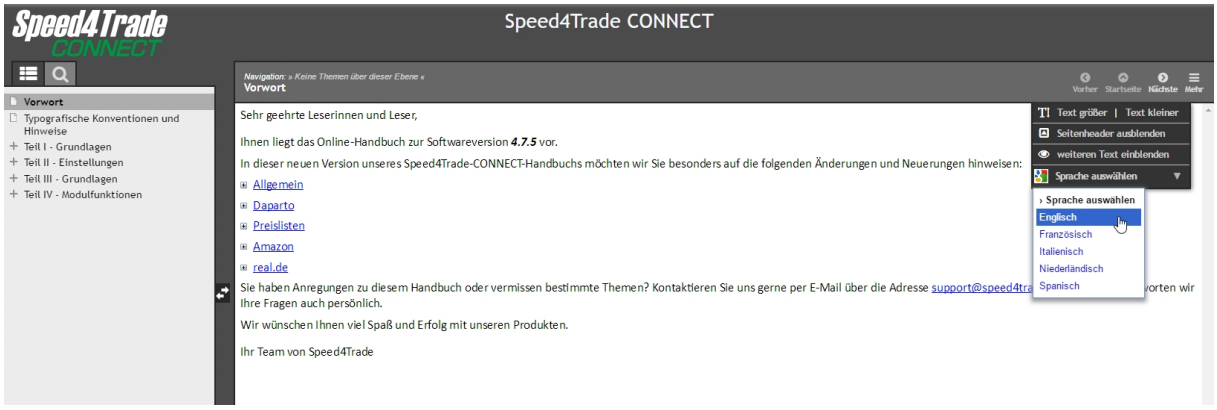

Illustration 7: Translation option for the Speed4Trade CONNECT Online Manual

Please use now the API documentation in addition to this guideline to continue in planning your implementation. (https://partner.speed4trade.com/api/dokumentation/index.html)

Below you will get a summary of preliminary API calls the most important basic settings.

Calls for pure query of basic settings**:**

- *fetchMandators* (clients)
- *fetchSubChannels* (sub - channels)
- *fetchAccounts* (accounts)
- *fetchTextModuleTypes* (text block types)
- *fetchImageModuleTypes* (faceplate types)

Calls to edit basic settings that can be platform - specific already preset:

- *fetchDeliveryChargeCategories* / *updateDeliveryChargeCategories* (Shipping cost categories)
- *fetchDeliveryTimes* / *updateDeliveryTimes* (delivery times)
- *fetchManufactures* / *updateManufactures* (manufacturer)
- fetchTemplates (listing templates)

Calls for the query job -related master data :

- *fetchPaymentMethods* (Payment)
- *fetchDispatchTypes* (delivery options)
- *fetchShippingCarriers* (shipping service)

5.1 Clients

Speed4Trade CONNECT Client path: Settings → User management → clients

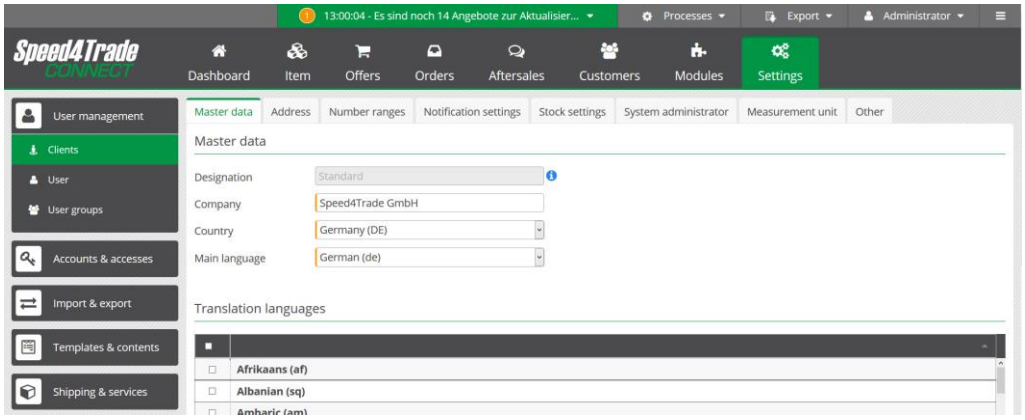

Illustration 8: Client - management

The Speed4Trade CONNECT is client -enabled. A "user" of the software can be assigned to several clients. At login, the user must decide for the client, with whom he wants to work. All data within the software are strictly separated according to clients. Conversely, this means that you must specify in each Webservice call for which client this data is to be set.

When clients are also the "translation languages " set, which here is defined, for which languages multilingual texts can be, for example, deposited with an article.

5.1.1 Webservice Call: fetchMandators (clients)

Standard

With the first function call must first be determined, the client data, as these are required for all further function calls. Thus, it is also ensured that all further calls also refer to the correct client.

At least the client IDs and the translation languages (ISO language code) for multilingual content (e.g. text blocks) must be read and processed. The clients and translation languages can be relatively rigid stored in the ERP system (e.g. in a configuration file / table), since they rarely change.

5.1.1.1 Languages

In addition to the main language can - if necessary - even more translation languages (e.g. English, French, etc.) can be specified in Speed4Trade CONNECT.

5.1.1.2 Currencies

The currencies for the item prices are directly connected with the country - specific sales channel (e.g. CHF at eBay.ch) and therefore need not be determined separately.

5.2 Sales channels

As sales channels, the parent market place names and shops are referred to; as Amazon or eBay. The supported sales channels are stored on the licensing firmly in the Speed4Trade CONNECT (and can not be changed via the Speed4Trade CONNECT client). However, since all actions on the country -specific pages, such as eBay.de or eBay.at , must be coordinated , can be used simultaneously with the corresponding "SubChannels". A call for "Channels" is not available nor necessary.

5.3 SubChannels

With which it is possible to set items in different countries - specific pages of sales channels. Some settings in the articles must be placed directly on a sales channel. For details, is explained in the article management.

To retrieve all currently supported sales channels, you need to call fetchSubChannels.

5.3.1 Webservice Call: fetchSubChannels (Sub - Channels)

Standard

The determined on fetchSubChannels subchannel IDs can be relatively rigid deposited in retail management, because it is often something changes here. However, it should be possible to simply add SubChannels (e.g. via a configuration file / table), if the customer wants to be active in other channels.

5.4 Accounts

Speed4Trade CONNECT path: Settings → Accounts & accesses

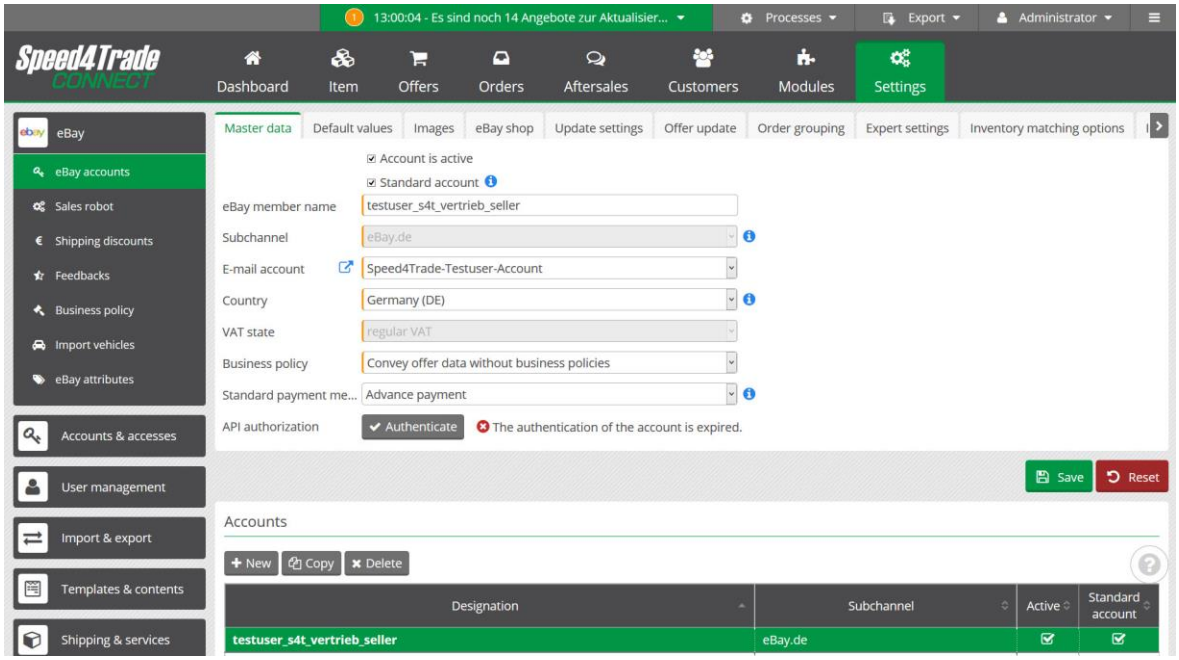

Illustration 9: Accountverwaltung

Depending on the rate, a different number of accounts per subchannel are maintained. For example, a customer created on eBay.de multiple sellers accounts and have in use. Similar to the distribution channels, it is also in the accounts so that certain settings to an article each account can be maintained, such as the price of the item with which you want to set a sales channel to this account this article.

5.4.1 Webservice Call: fetchAccounts (accounts)

Standard

Again, the determined via fetchAccounts account IDs of the market places can be relatively rigid deposited in retail management, because it is often something changes here. However, it should be possible to add additional accounts (e.g. via a configuration file / table), if the customer wants to use other accounts.

5.5 Text block types

Speed4Trade CONNECT path: Settings → Templates & contents → Text block types

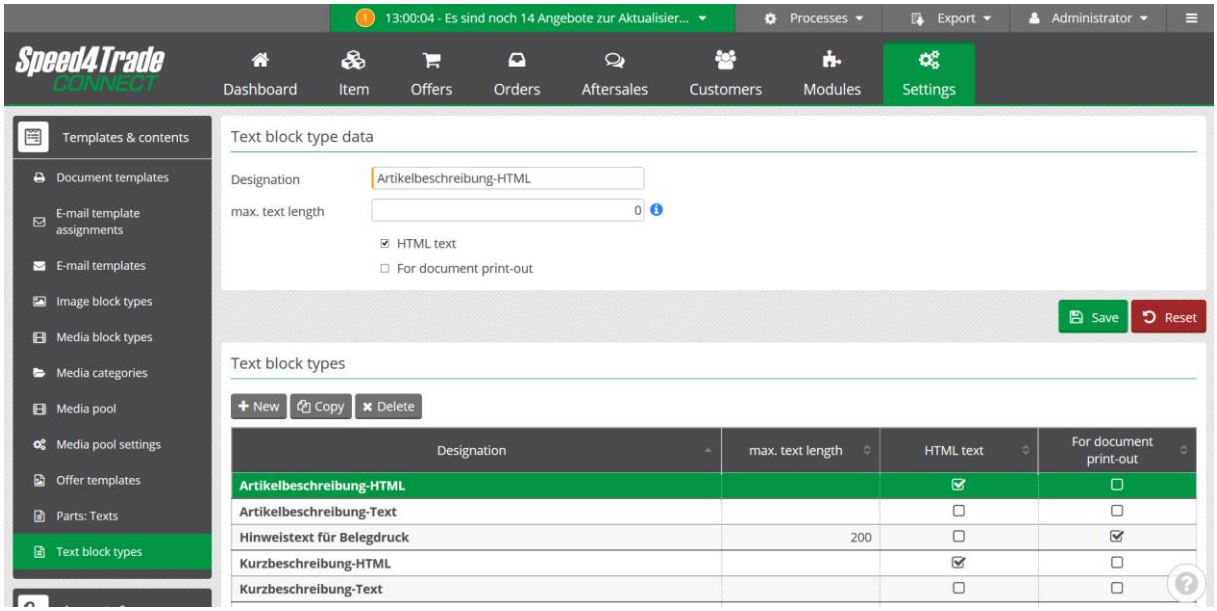

Illustration 10: text block types

Speed4Trade CONNECT works when generating the offer content and descriptions with text blocks. This allows for maximum flexibility. About different text modules, specially adapted to the requirements of the market places, descriptions can be created as needed per item.

The number of the description fields for an item is not fixed, i.e. every user can decide how many fields he needs for the presentation of its products. But this in turn means that at the delivery of the item from the external system in the Webservice call must specify the text to which block type, for example, the long text to be set.

5.5.1 Webservice Call: fetchTextModuleTypes (text block types)

Standard

The two text block types can be distinguished HTML and Plain - Text:

• Required text block types:
- text modules for HTML text ◊ for eBay and shop
- text modules for limited HTML ◊ Amazon My package, Manuals, Rakuten
- text blocks for Plain - Text ◊ for all channels

• Possible additional text blocks (examples)
- Short description (html text)
- Long description (HTML text below)
- Brief Description (Plain Text)
- Long Description (Plain Text)
- Technical data (HTML text below)
- meta data page Title (plain text)
- meta- data description (plain text)
- meta-data tags (Plain Text)

The text block types are first defined in Speed4Trade CONNECT. The maintenance of the values for example, short and long description is in retail management. It must be created between the texts from the ERP system and the text block types a mapping. This is done via the IDs of each text module in retail management.

Note!
Are multi-language article text desired, and these must be maintained in another language-dependent text blocks. The mapping of the various fields with the different language texts must then be made at the presentation of the texts in the article on the basis of a Speed4Trade CONNECT ISO language codes.

5.6 Image block types

Speed4Trade CONNECT path: Settings → Templates & contents → Image block types

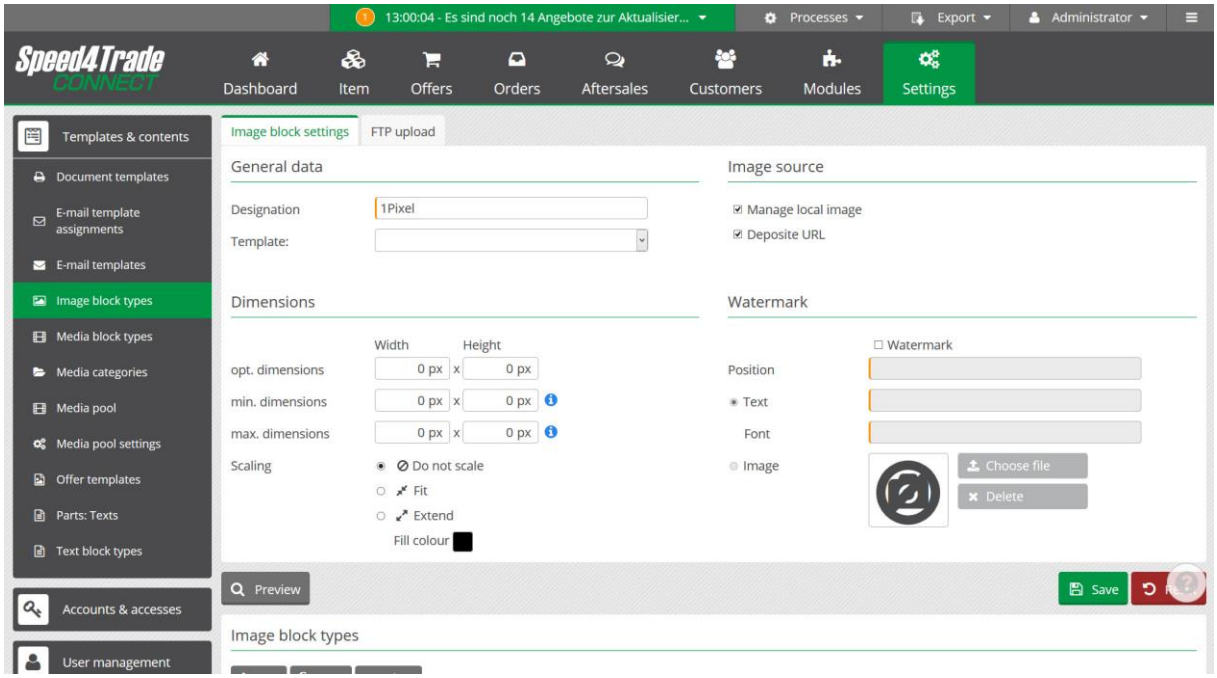

Illustration 11: faceplate types

Similar to the text block types it is the same with the faceplates, i.e. the user decide how many images he wants to deposit them with the articles. About the types can very flexibly different - be deposited per item images - possibly also market place - specific. Images are always scaled based on the stored configuration.

This also leads in turn to the fact that during the transfer of articles from the external system in the Webservice call must be specified composition on which block type, for example, the main image is to be placed.

5.6.1 Webservice Call: fetchImageModuleTypes (image block types)

Standard

The 2 image block types can be distinguished UNC and URL image:

- Required faceplate types:
  - UNC images / local images ◊ example eBay and shop
  - URL images ◊ example Amazon , eBay and Manuals , My package Rakuten

- Possible faceplates (examples)
  - Smaller image for gallery view
  - Large image with maximum resolution for zoom feature
  - Specially adapted images per sales channel (eBay, Amazon, etc.)

The image block types are first defined in Speed4Trade CONNECT. Maintenance is done on the image functionalities in retail management. It must be created between the images from the ERP system and the faceplates of a mapping. This is done via the ID each faceplate in the commodity economy. The images should ideally already in retail management in the correct resolution (greater than 600 pixels) and quality be stored and transferred .

Note:
URL images are preferred because they can be used by all sales channels. Local images are possible only on eBay and shops. Depending on the sales channel can be used per item up to 12 images. UNC paths are implemented nationally and no URL pictures.

Optional local images can be provided via the built in Speed4Trade CONNECT photo upload to an FTP account as URL pictures.

If you pass images as Base64, it can with large datasets (partly already from 2 MB per request) cause problems, because the standard setting in the TomCat will not accept requests on this scale.

5.7 Media Management

For example, PDF documents and product videos assigned to an item and this can be uploaded to the respective marketplaces and online shops, a media management is integrated into Speed4Trade CONNECT.

The assignment of the media to the articles can currently only be done in Speed4Trade CONNECT and therefore can not be set by the commodity economy on the Webservice. This functionality is the future but still extended.

5.8 Application of image, text and media modules in offers

The maintainable in the basic settings of the software image, text and media block types get over the care of the image, text and media files in the article, the article-specific data range. The following graphic illustrates the basic logic blocks, wildcards, articles and offers. It should be noted that there are fundamental differences in this system, depending on the selected sales platform. Please refer to the specifics later in this section.

Basically: To submit a quote always an item must be available. It represents the central object from which a quotation can be prepared.

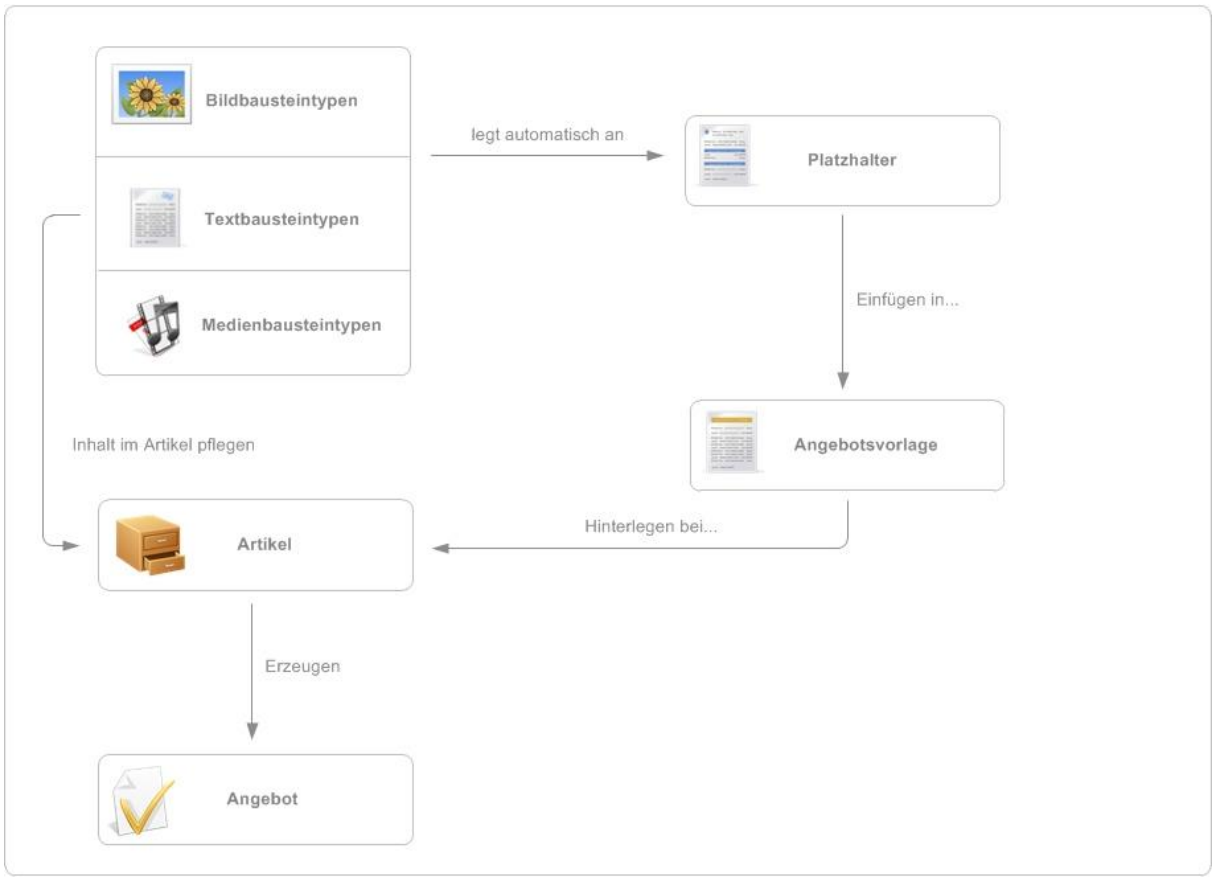

Illustration 12: Application of image, text and media modules in offers

Notes:

- If a text, image or media block type is created, it is available from the date of creation in the article for the care.
- If a text, image or media block type is created, the software automatically creates a placeholder for this purpose, which in turn can be inserted in the listing templates.
- The template and thus the embedded placeholder there are selected items in the appropriate sales channel.

- When creating / preparing the bid, the placeholders are replaced by the text or image data. This place is the content of each block in Article empty, so is left blank in the generated offer without spending the placeholder.

5.9 Centralized Attribute Management (CAM)

Standard

With version 3.5.37 Speed4Trade CONNECT first time the CAM was introduced. Each distribution channel is slightly different. We therefore want to reduce the complexity by implementing a sales-channel-independency introducing a centralized data management. These three steps are necessary in the future:

- Create attributes and their values (master data)
- Matching to the platform-specific attributes and values
- Attribute values to assign to the item
-

By default, the CAM is to be used in Speed4Trade CONNECT (or masks on the Mashup embed) . As advanced feature these three steps using the Speed4Trade CONNECT API directly from the external system (for example, PIM) can be addressed.

New sales channels such Hitmeister, Locafox and Websale been based solely on this new system. A complete sales channel -independent interface is achieved if in the future also a "price lists function" is implemented.

Note:
The CAM cannot be mixed with the previous approach. That means if a sales channel uses the CAM in a call, all other sales channels have also to use the CAM.

St
Speed4Trade CONNECT Pfad: Settings → Labeling → Attributes

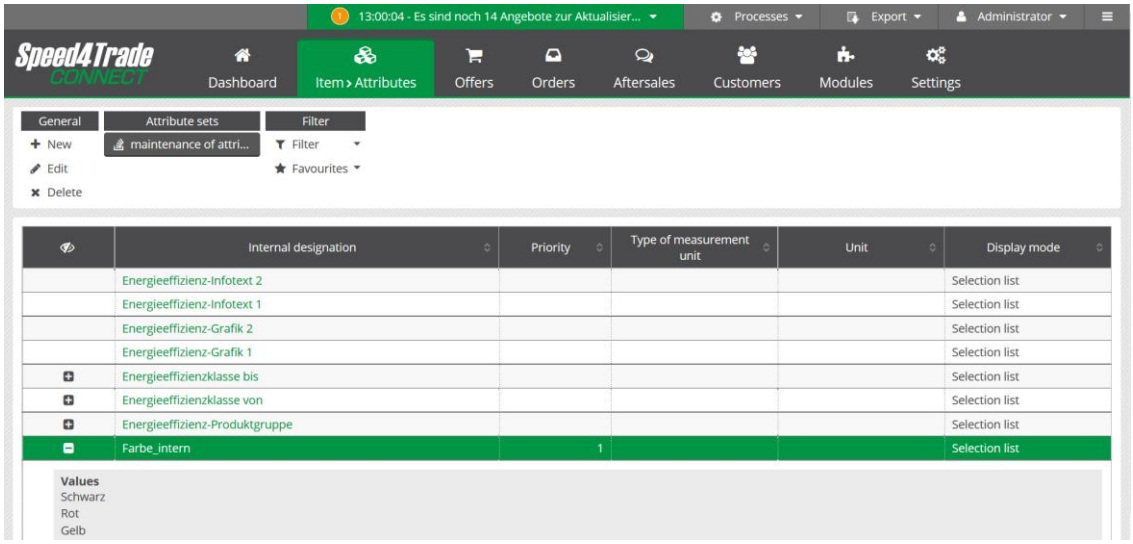

Illustration 13: Item Attributes

Below you can see an example at a glance how the Centralized attribute management can be integrated. The assignment of attributes to product is taken up in Section 7.2 again.

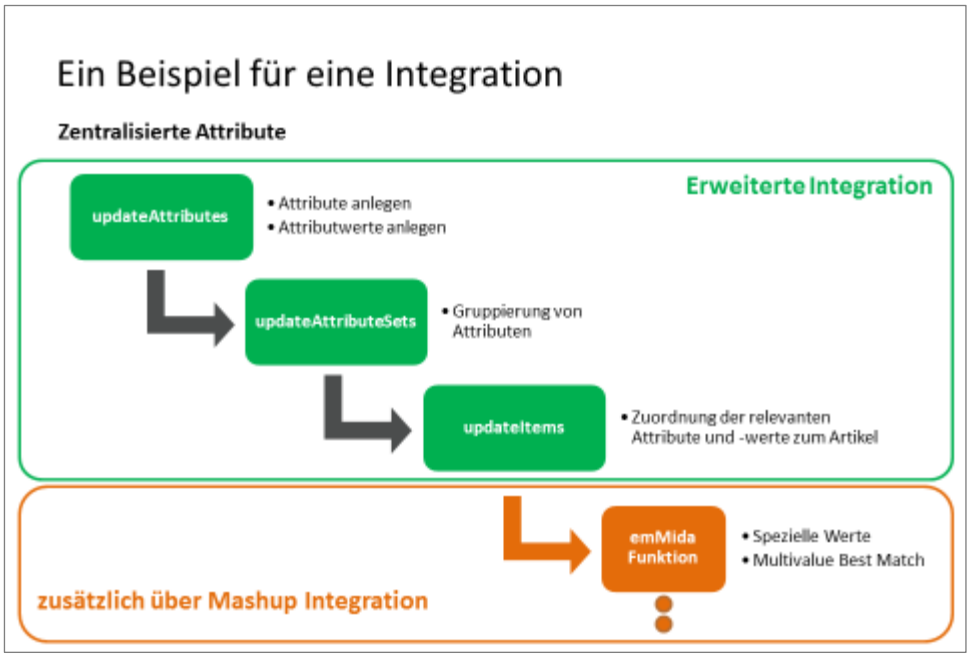

Illustration 14: Integration samples for Attributes

5.9.1 Webservice-Call: fetch / updateAttributes (central attributes and values)

- maintenance of and creation of new attributes on new call update attributes
- getting existing attributes with new call fetchAttributes for initial data migration
- avoid large number of predefined of attribute values: use free text attributes
- sequence is defined when created new , can be modified via GUI
- attribute_identifier and attribute_id serve as key for further calls

5.9.2 Webservice-Call: fetch / updateAttributeSets (central attribute groupings)

- in Attribute Sets You can group attributes in sets .
- these sets can be selected in the article and then predisplayed before the grouped attributes in products.
- attribute mapping of attribute names or Speed4Trade CONNECT internal ID

5.10 Pricelists (updatePricelist)

Standard

With the version Speed4Trade CONNECT 4.6.46, price lists were introduced for the first time. Each sales channel is slightly different. The implementation of price lists is necessary for a sales channel independent interface and is mandatory for future certified solution.

- Collection of price lists and prices (data transfers or reconciliation) – fetchPricelist
- Create price lists and maintain prices

The goal is to implement new sales channels in Speed4Trade CONNECT immediately in your system. For this, it must be possible in your implementation to create a new price list in ERP and to maintain sales prices. The new price list must be able to be assigned by configuration only (without programming your interface).

Hint:
When changing to new price lists, please note:

- Mixing of price functions is possible, e.g. use the price list for eBay, Amazon and Speed4Trade COMMERCE, for the other channels the previous methods
- Price list prices can no longer be maintained in the interface of Speed4Trade CONNECT. Prices must be imported via the API or via CSV files a BPL. Thus, the application of the mashup integration model follows.
- Other prices, such as; Customer group and season prices, strike prices, basic prices, etc. are still shown as before.

Prices for new incoming sales channels can only be defined via the price list function.

Speed4Trade CONNECT path: settings → master data → pricelists

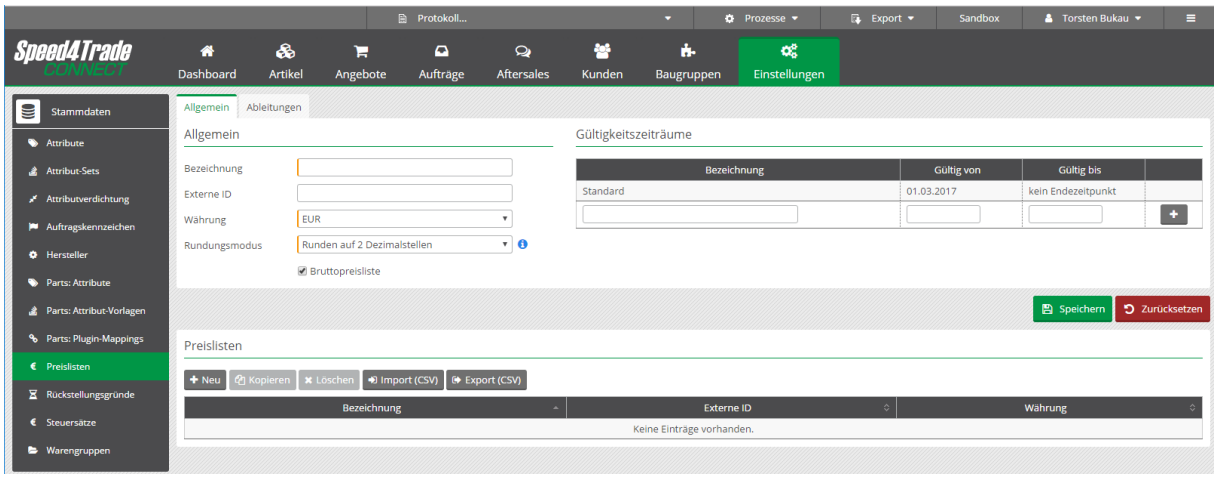

Illustration 15: Pricelists

In Speed4Trade CONNECT, the function Basic Price Lists (BPL) and Derived Price Lists (DPL) are available.

The following is a sample of a standard integration using Basic pricelists (BPL)

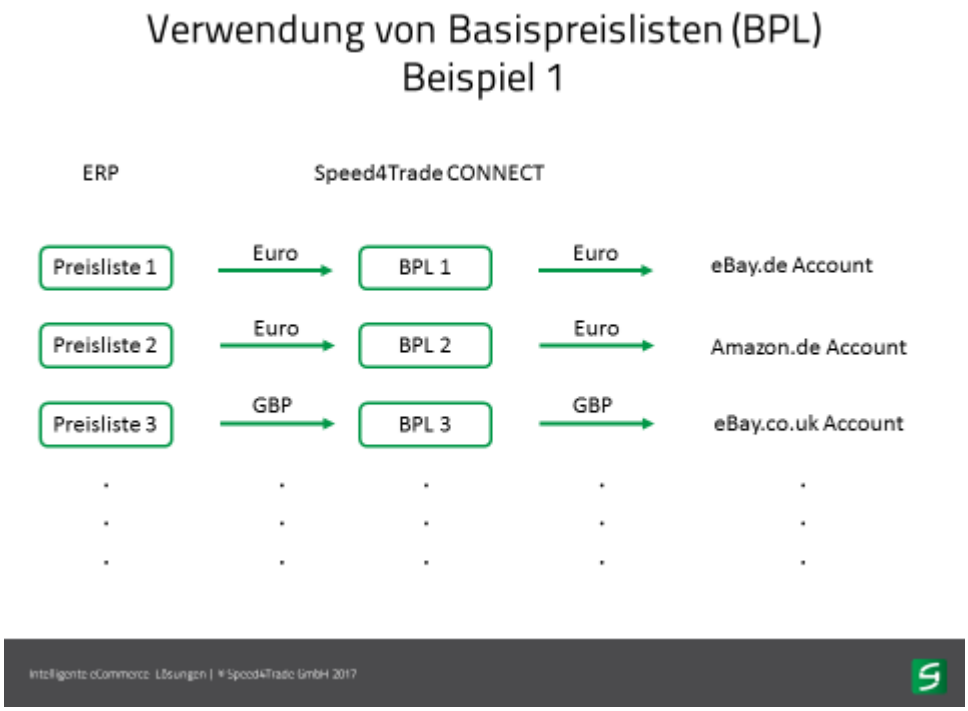

Illustration 16: Usage scenario of Basic Price Lists (BPL)

If the external system cannot transfer several sales prices for an item, we can show you further possibilities for implementation in the integration workshop or upon request.

5.11 Return Categories

Speed4Trade CONNECT path: Settings → Shipping & services → shipping cost categories

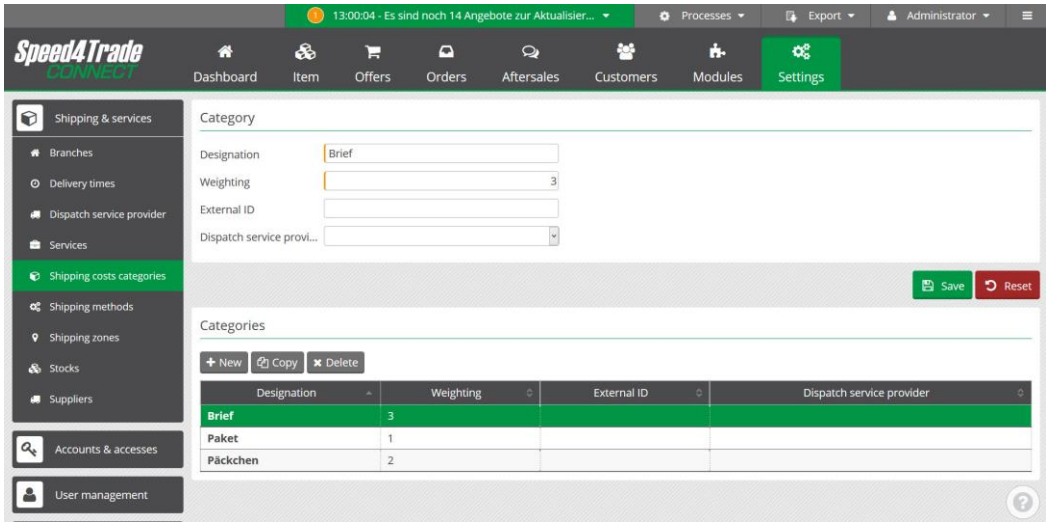

Illustration 17: return categories

5.11.1 Webservice Call: fetchDeliveryChargeCategories / updateDeliveryChargeCategories (Shipping cost categories)

Standard

Shipping cost category is a peculiarity of the marketplace eBay to define the shipping cost. The shipping cost categories can be detected in Speed4Trade CONNECT. Thus, applying through the ERP (updateDeliveryChargeCategories) is not absolutely necessary. The information presented in Speed4Trade CONNECT shipping cost categories should, however, be read with fetchDeliveryChargeCategories in order to be able to assign the articles that are to be set on eBay, in a later step. It is enough to store the data field *"delivery_charge_category_id".*

5.11.1.1 Returns

The other sales channels Returns are usually set directly in the backend of the marketplace.

5.12 Delivery Times

Speed4Trade CONNECT path: Settings → Shipping & services → Delivery Times

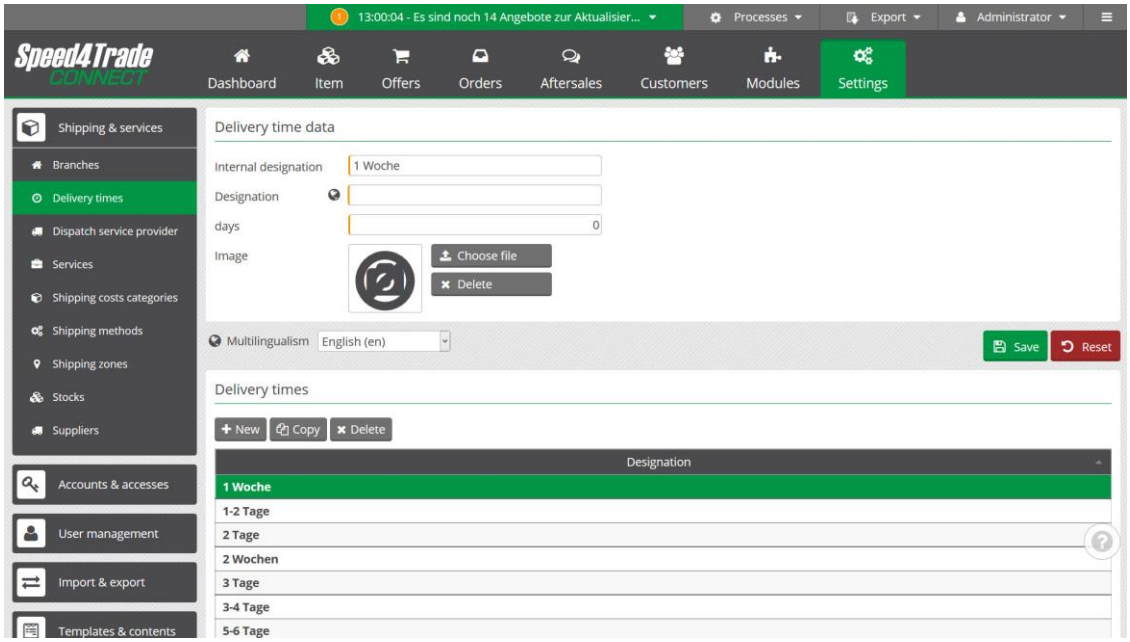

Illustration 18: Delivery Times

5.12.1 Webservice Call: fetchDeliveryTimes / updateDeliveryTimes (delivery times

Standard

The definition of delivery times is especially important for Amazon and shop links. If, for example, specified at Amazon no delivery time, Amazon will automatically turn off by a delivery time of 1-2 days.

Delivery times are normally maintained in Speed4Trade CONNECT. But if they are already in retail management exist, they can also be set directly via the Webservice Call *updateDeliveryTimes* and read again by *fetchDeliveryTimes*.

If necessary, a mapping between ERP and Speed4Trade CONNECT delivery times must be carried out. Once this is done, the delivery times when you submit the article on *updateItems-> delivery_time_id / delivery_time_identifier* can be set directly.

5.13 Manufacturer

Speed4Trade CONNECT path: Settings → labeling → manufacturer

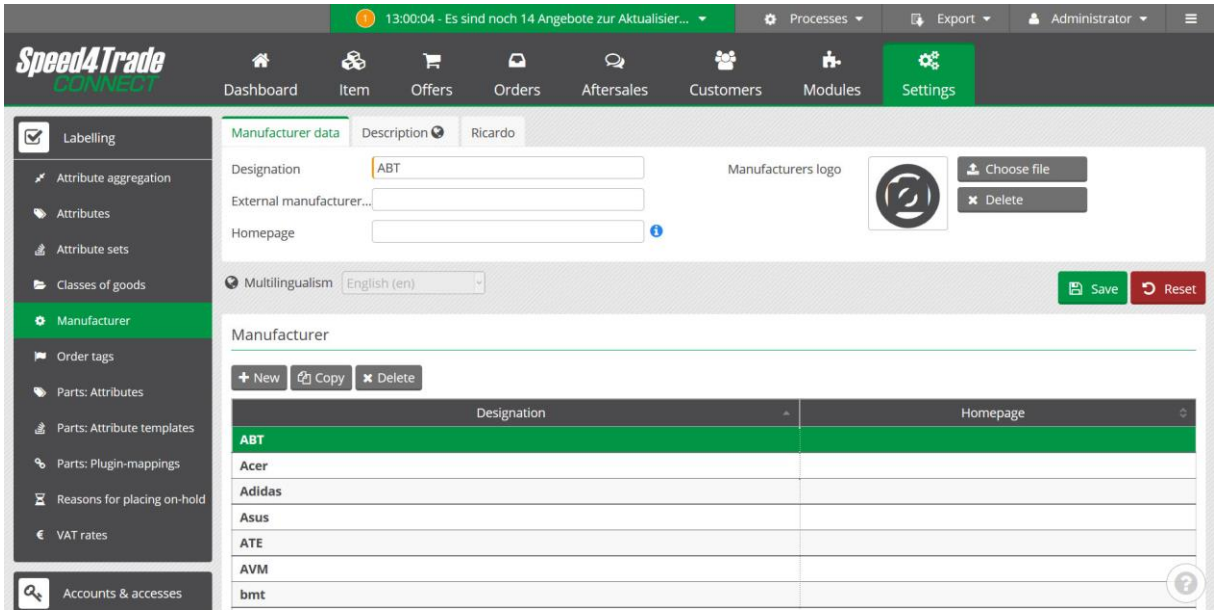

Illustration 19: Manufacturer

5.13.1 Webservice Call: fetchManufacturers / updateManufacturers (manufacturer

Standard

The indication of a manufacturer is in certain categories on certain sales channels a required field. We, the manufacturer often used as search and filter criteria. Therefore, the maintenance of a manufacturer's products for increased sales opportunities.

The manufacturer may be invested in Speed4Trade CONNECT. If they are already stored in the ERP system, it makes sense to pass on the Webservice with the Call *updateManufacturers* to Speed4Trade CONNECT. As a result, the manufacturer ID are set read about *fetchManufacturers* from Speed4Trade CONNECT and *vendor_id* by *updateItems* in the field in the item again.

5.14 Warehouse / Stock

Speed4Trade CONNECT path: Settings → Shipping & services → stocks

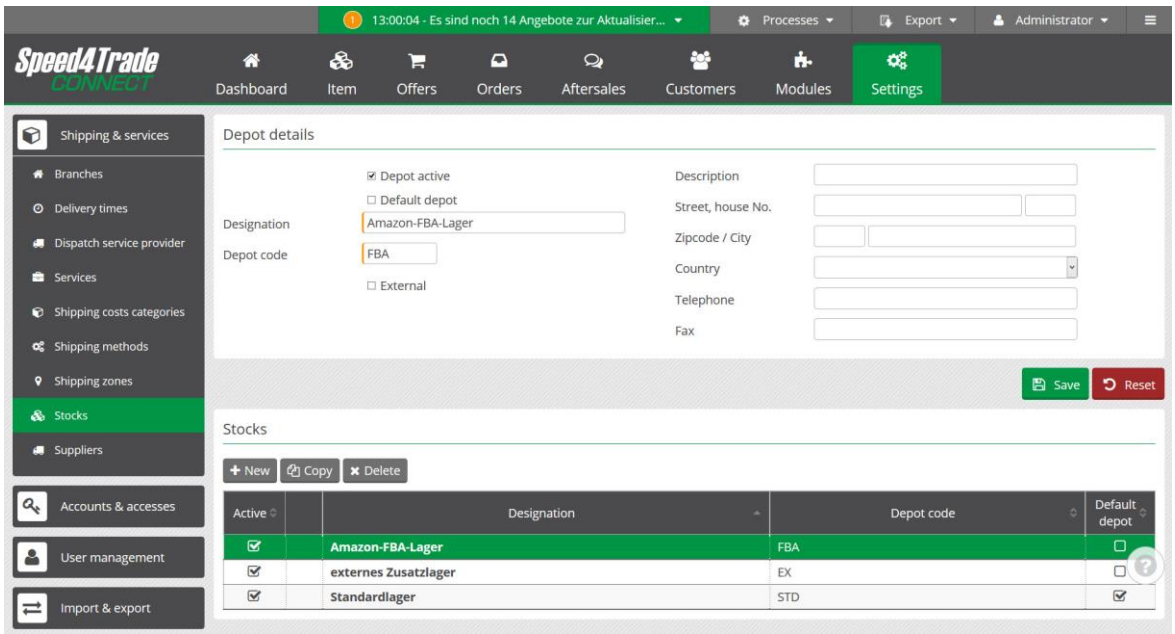

Illustration 20: Stock

5.14.1 Webservice Call: fetchWarehouses (stock)

Standard

The Speed4Trade CONNECT is more storage capability. That is, whenever stocks over *updateItem* or update stock count be booked, the corresponding bearing in the field is with *warehouse_id* specify. Therefore, it is useful to read the available storage in advance and save between.

6 Product Groups

The definition of product groups is a very efficient method to easily match large amounts of similar products to the market place categories. The structure of departments essentially corresponds to the structure of departments in retail management. This product groups can be structured as a tree of any depth. Since commodity groups are usually a fundamental part of the ERP, this should serve as a basis for the Departments of Speed4Trade CONNECT.

The commodity groups should therefore be applied by the external system in Speed4Trade CONNECT while also correspond to the structure of the external system. Thus, the user will find in Speed4Trade CONNECT easy to navigate in its product range.

Each article must always be a marketplace and / or shop category. To this end, the department from the ERP in Speed4Trade CONNECT are "mapped" to the categories of market places and / or shop. An article can only ever be assigned to a department.

Note!
The product groups do not yet directly the categories of a marketplace or online shops. The assignment is produced only by the mapping within the Speed4Trade CONNECT.

The product area shall meet the following additional convenience functions in Speed4Trade CONNECT:

- filtering of articles , that is, when you select a group , only the assigned articles are displayed. Thus, the range of articles can be well structured.
- inheritance of properties , that is, in a department can deposit e.g. standard text modules or standard faceplates . Each item in this product group gets automatically inherited these values. Inheritance can be , for example, be used if it is in the group to a particular manufacturer, for each article always a manufacturer image is to be defined along with description.

The management of departments in Speed4Trade CONNECT can be contacted as follows:

Speed4Trade CONNECT path: Settings → labeling → classes of goods

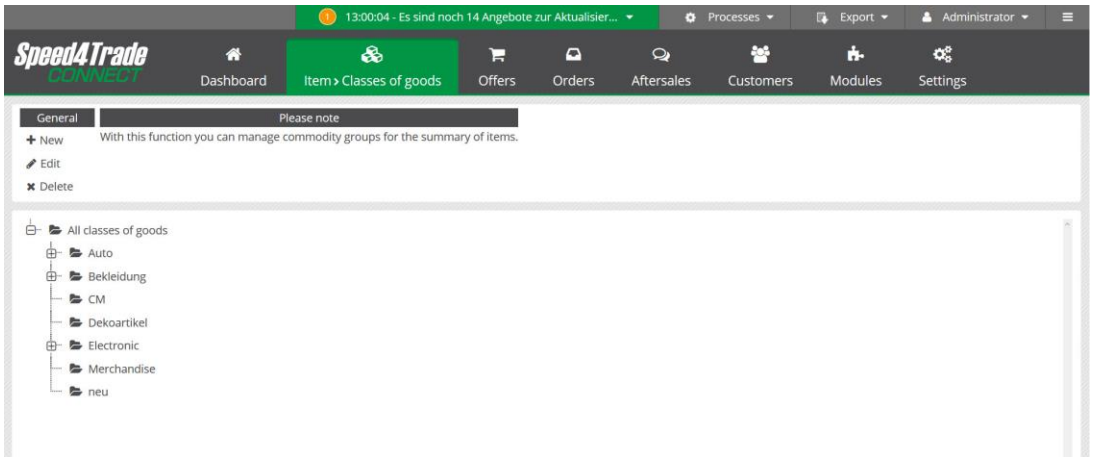

Illustration 21: classes of goods

6.1 Webservice Call: UpdateCategories / fetchCategories (departments)

Komfort

About the calls the department can be created (*UpdateCategories*) and will read again (*fetchCategories*) . This can be used later in the article transfer both the department ID (*EXTERNAL_ID*) of commodity economy and the internal Speed4Trade CONNECT goods group ID (*category_id*) .

Required preparatory Calls:

- *fetchMandators* (clients)
- *fetchTextModuleTypes* (text block types)
- *fetchImageModuleTypes* (faceplate types)

7 Item master data

In the article master both the basic data that are the same for all sales channels, as well as the sales channel - specific data is maintained. In the following chapter examples received only on eBay and Amazon. The other market places and shops, please see the documentation.

Required preparatory Calls:

- *fetchMandators* (clients)
- *fetchTextModuleTypes* (text block types)
- *fetchImageModuleTypes* (faceplate types)

Additional calls to preset more basic data:

- *fetchDeliveryChargeCategories* (Shipping cost categories)
- *fetchDeliveryTimes* (delivery times)
- *fetchManufactures* (manufacturer)

7.1 Item Basic data

I n addition to the preparatory calls and a reference for each article should be set to the previously created group to use the mentioned convenience functions (see fields *category_id* and *external_category_id* in Call *updateItem*).

Note!
Some property values are automatically used for sales channels. Thus, for example, the internal item number is passed to eBay, the EAN code and the internal article number to Amazon, etc.

7.1.1 Basic data - general

Speed4Trade CONNECT *path: item → Basic data → general*

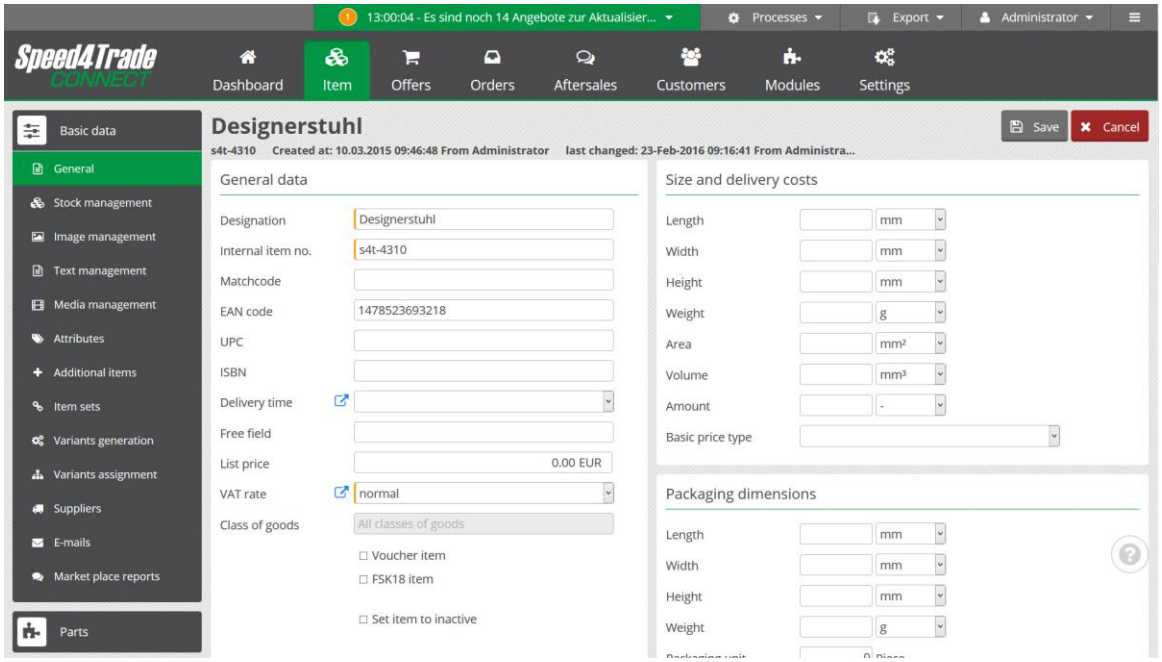

Illustration 22: item basic data

7.1.2 Basic Data - Text Management

Speed4Trade CONNECT path: item → Basic data → text management

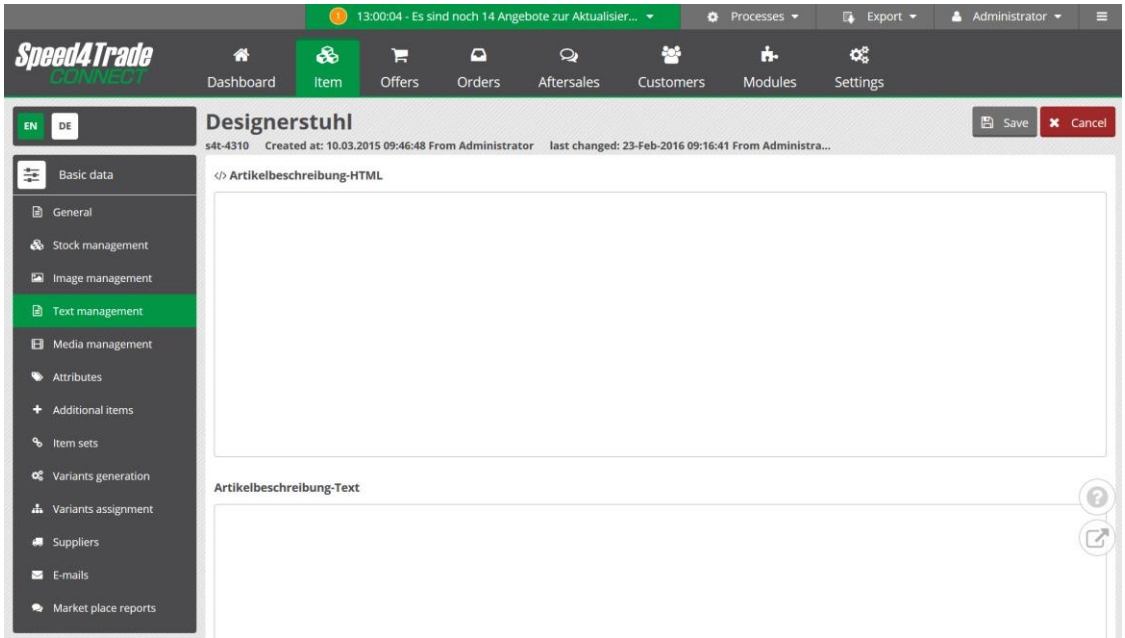

Illustration 23: item text management

7.1.3 Basic Data - Image Management

Depending on the setting, a local image or a URL per faceplate type to be set.

Speed4Trade CONNECT path: item → basic data → Images Management

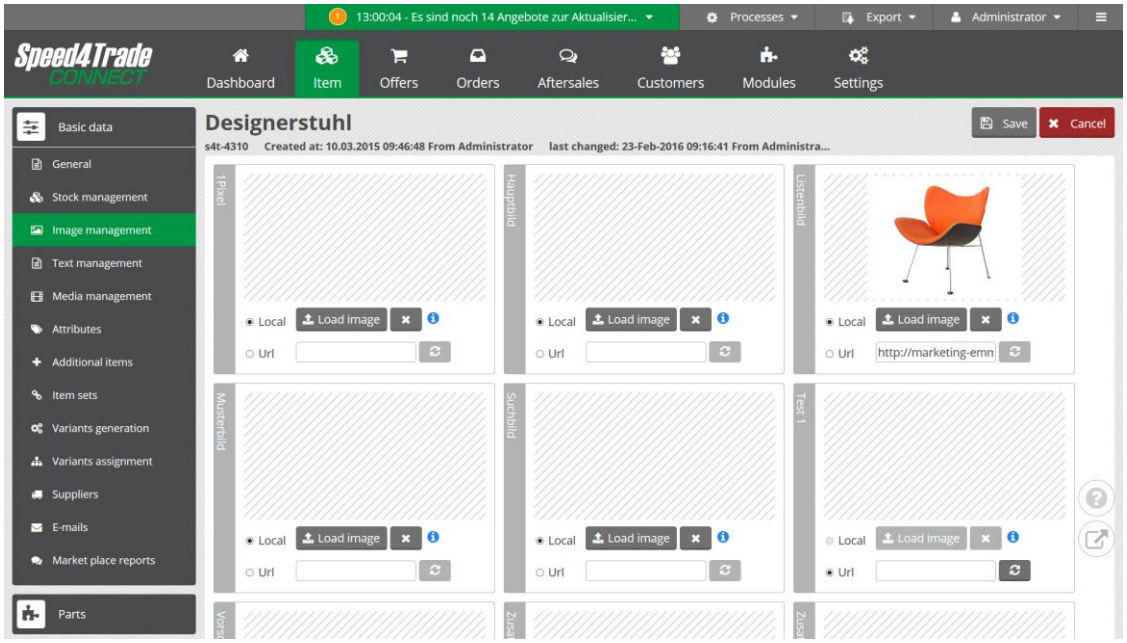

Illustration 24: item Images Management

7.2 Webservice Call: update items (item master data)

Standard

Below is a list of basic fields in the Item Master, which should be placed on *updateItems*. For a complete list directly in the Webservice API online documentation for the call *updateItems*. Please note there also the lists of values and notes at the bottom of this page for the corresponding function call.

The "*significant_key*" field specifies which field is used for identification of the article during the update. This may be a record number "*item_id*" or an internal item number be "*internal_item_number*". If no number is specified, a new record, that is created article.

7.2.1 Article -> Basic Data -> General

Speed4Trade CONNECT field name	description	Field name WebService
Description	Required field	identifier
internal item number	Required field	internal_item_number
matchcode	For example. Search for products	match_code
EAN	Important for Amazon, eBay an Google Produc Search. (principle there should be no article without EAN number)	ean_code
Manufacturer Part Number	Article number of the manufacturer	vendor_item_number
Manufacturer	Manufacturer ID	vendor_id (manufacturer has already invested in Speed4Trade CONNECT, ID was determined via Webservice call = *fetchManufacturers*) external_vendor_id (manufacturer is new about ERP in Speed4Trade CONNECT = *updateManufacturers*)
Delivery time	Important for Store an Amazon (nursing can take place both in retail managementan in Speed4Trade CONNECT)	delivery_time_id (Delivery time was invested in Speed4Trade CONNECT, ID was via Webservice call = *fetchDeliveryTimes*) delivery_time_identifier (Delivery time is applied via ERP new in Speed4Trade CONNECT)
Free field	Internal filter criterion	variable_field
List price	List price of the item Only for information purpose – no functional usage	list_price
Vat-rate	Required field:	tax_type_id Fixed values:

	Surrender of the relevant VAT rate ID from the ERP
	• tax rates:
	  ○ normal
	  ○ tax-free
	  ○ decreased
		1 = free
		2 = SALE
		3 = normal

Product group	Can be set when you created the article, otherwise no assignment	category_id (Category was created in Speed4Trade CONNECT, ID was determined via Webservice call = *fetchCategories*) external_category_id (was newly created by ERP in Speed4Trade CONNECT = *updateCategories*)
Rated 18-item	Can be set when FSK items are sold. Relevant for shops.	is_adult_product
Set to inactive products	Note: The setting of this flag only has an effect at stores. There are thus offers not automatically terminated.	active
Item measurements	Should be set at appropriate articles, as many customers ask for it • Lenght / unit • Width / unit • Height / unit • Weight / unit • Area / unit • Volume / unit	length / distance_unit_id width / distance_unit_id height / distance_unit_id weight / weight_unit_id area / area_unit_id volume / volume_unit_id
Base price type	Base price type of article. Relevant for basic price paid article (eg Unit Price for 100ml for drinks)	base_price_type_id If you specify the base price typs of corresponding unit value must be set.
Packaging dimensions	Should be set at appropriate articles. • Lenght / unit • Width / unit • Height / unit • Weight / unit	package_length / package_distance_unit_id package_width / package_distance_unit_id package_height / package_distance_unit_id package_weight / package_weight_unit_id
Shipping cost categories	Shipping cost categories are important for eBay. The associated costs to be maintained in Speed4Trade CONNECT. The shipping cost categories (ID) shall then be placed on the commodity economy.	delivery_charge_category_id

7.2.2 Item -> Basic Data -> Additional Articles

Speed4Trade CONNECT field name	Description	Field name Webservice
Additional articles	Currently only for shops and Manuals possible on Amazon a recommendation is passed.	cross_selling_item

7.2.3 Item -> Basic Data -> article-bundle

Speed4Trade CONNECT field name	Description	Field name Webservice
Article-bundle	Speed4Trade CONNECT can deal with Set-Artikeln/Bundles. It should be clarified in advance, always check Speed4Trade CONNECT must "know" whether it is a set items, or if only one item is passed and the goods economy everything, for example, inventory management, for set items / Bundles / BOMs takes.	item_set.

7.2.4 Article variants

The Speed4Trade CONNECT also controls the handling of variations. The necessary for the variants of properties (color, size, etc.) and associated values (eg, red, blue, or M, L, XL, etc.) to be defined.
The generation of variants can be performed directly variant via the Call updateItems under the wrapper element. In addition to the variant sellable items (eg T-shirt in blue color and size L) also called SKU (Stock Keeping Unit), also Main article must always be a (not purchasable) are generated that groups or summarizes the variants logical. The main article can be marked as such variant_standard with the field. First, the main products in a first updateItems call must be passed. In a second updateItems call then the variants can be assigned.

In addition, it is checked in updateItems call, whether the properties / value combination already exists in the master data. If this is not the case, the combination is created.

7.2.5 Item Attributes (Centralized Attribute Management - CAM)

Allocation of central attribute values item_attributes the item on the Call updateItems under the wrapper element .

Notes :
An exception are the variant-creating attributes that continue to be allocated through variant / attributes . A new system of attributes and attribute values incl . Marketplace data is on this call also

possible , but in no way suitable best practice and not for large amounts of data . This should be done on update attributes .

7.3 Webservice-Call: fetchItems

Having now explained in detail how the item master data can be passed to Speed4Trade CONNECT, it should be mentioned briefly that using the call fetchItems the product data can be collected. This is especially useful if you connect to an existing Speed4Trade CONNECT installation. All existing item master data must be imported into the ERP initially. This initial re-import works via a CSV export or alternatively through the call fetchItems.

8 Sales channel - specific data

Each sales channel has different settings, which in turn are also different from the complexities. It should be placed as much data as possible about the external system to keep the maintenance effort in Speed4Trade CONNECT as low as possible. Setting a sales channel - specific data belongs to the ' Advanced ' implementation. The exception is the passing of title and sale price per channel.

8.1 Item - Description and Price

Standard

Why is the setting of title and price per sales channel via the ERP system so important? The selling price and the name are among the most important fields per sales channel. Thus, the products most differ. Especially the price should be centered in the main system, will speak of commodity economy maintained and managed. Furthermore, these two fields cannot be set via the Speed4Trade CONNECT comfort functions templates, inheritance, and multiple editing. That is: Can the item price and description are already set on the external system, which is a considerably easier for the dealer.

Speed4Trade CONNECT field name	Description	Field name Webservice
Item - Description	Required field	title
Standard price	Standard price for Amazon	price[1]
Starting price	Starting price for eBay	minimal_bid
Buy it now price	Buy it now price for eBay	buy_it_now_price

[1] Besser: Preislisten verwenden!

8.2 Amazon marketplace

Speed4Trade CONNECT path: Item → Amazon → general

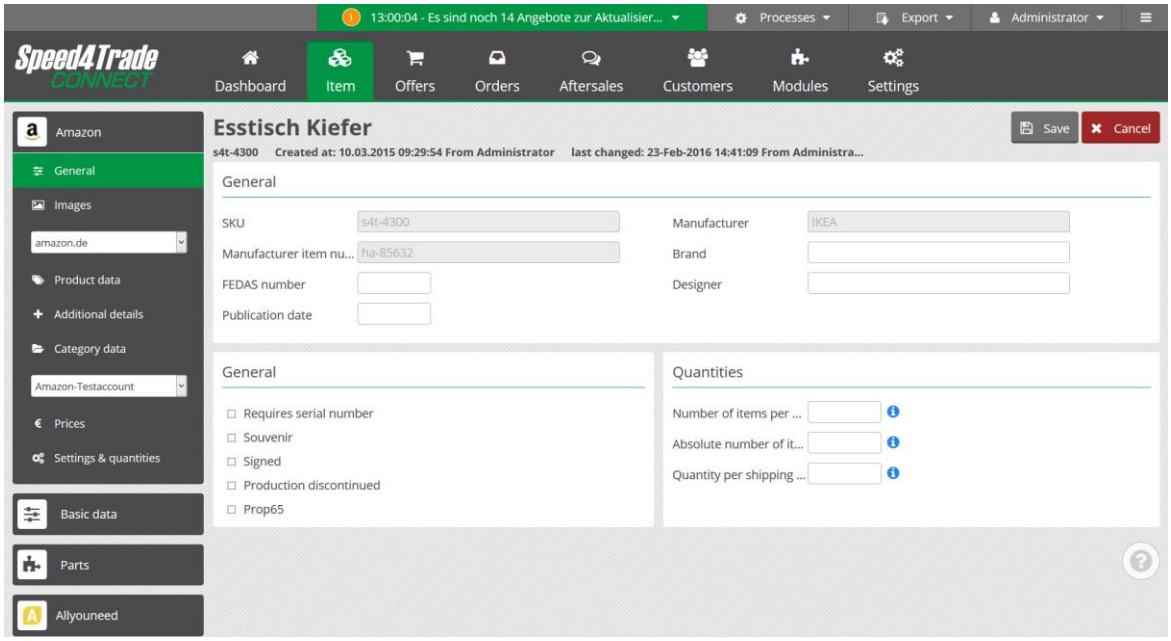

Illustration 25: Item data - Amazon

At Amazon, there are data that generally apply to all countries Platforms Amazon. This includes, for example, the manufacturer or the SKU = internal part numbers. Furthermore, there are settings that apply only to the respective national Amazon platform - in this case, for example, for amazon.de - can be set. Finally, there are values that can be set per Amazon account, such as the price.

Required preparatory Calls:

- *fetchSubChannels* (Sub-Channels)
- *fetchAccounts* (Accounts)

8.2.1 Webservice Call: updateItems (Amazon -specific data)

Compared to other online marketplaces, there are certain peculiarities on Amazon. At Amazon, there is , for example, so that a so-called product catalog will be made available . If the item which is to be set at Amazon, has been listed by another seller, so that already the item information in the product catalog are available , then only the following data must be maintained in the Speed4Trade CONNECT.

- Manufacturer
- Description
- Inventory
- Price (better to use pricelists!)
- State
- EAN / ASIN-Number

If the article which is to be set , never before been listed at Amazon, in addition to the points listed above , additional data must be.

8.2.1.1 Item -> Amazon -> general

Speed4Trade CONNECT field name	Description	Field name Webservice
SKU	Is taken from the product data base (internal item number)	internal_item_number
Manufacturer	Is taken from the product data base	vendor_id
Manufacturer part number	Manufacturer is taken from the product data base	vendor_item_number

In the "Amazon -> General" No data needs to be set by the commodity economy, as these are covered by the basic data. For the SKU number, the stock number is used.

8.2.1.2 Item -> Amazon -> Images

Speed4Trade CONNECT field name	Description	Field name Webservice
Main picture	Faceplate-ID	main_image (optional)
Sample image	Faceplate-ID	swatch_image
Search image	Faceplate-ID	search_image
Additional picture 1	Faceplate-ID	pt1_image
Additional picture 2	Faceplate-ID	pt2_image
Additional picture 3	Faceplate-ID	pt3_image
Additional picture 4	Faceplate-ID	pt4_image
Additional picture 5	Faceplate-ID	pt5_image
Additional picture 6	Faceplate-ID	pt6_image
Additional picture 7	Faceplate-ID	pt7_image
Additional picture 8	Faceplate-ID	pt8_image

The images, which are to be used for the marketplace Amazon, must, among other things meet the following requirements:

- isolated images → no background
- minimum resolution 1024 x 1024px
- no watermark may contain

Note!
If only item on Amazon to set which are already present in the product catalog, the Leave of the faceplates in the tab "Amazon" not mandatory.

8.2.1.3 Item -> Amazon -> product data

Country-specific data set, for example, for Amazon.de.

Required preparatory Calls:

- fetchTemplates (listing templates)

Speed4Trade CONNECT field name	Description	Field name Webservice
Item description	Required field	title
Point 1	If possible as promotional	bulletpoint1
Point 2	If possible as promotional	bulletpoint2
Point 3	If possible as promotional	bulletpoint3
Point 4	If possible as promotional	bulletpoint4
Point 5	If possible as promotional	bulletpoint5
Description	Assigning a listing template. Can be done via multiple editing in Speed4Trade CONNECT, or one gets the offer templates in advance via fetchTemplates from.	description_template_id
Standard product number	Required field EAN number is taken from the product data base ASIN number could - be set - if known.	standard_product_id
Standard product number type	Specifies which part number is used (EAN, ASIN, ISBN, etc.)	standard_product_id_type

For adjusting to foreign language sales channels the article names and descriptions are also multilingual maintain.

The care of the bullet points is only relevant if the Amazon account is entitled to re-create or overwrite information in the product catalog. The same applies to the care of the "Additional Information" in the Amazon market place-specific product data.

The care of the ASIN and EAN number is usually in retail management. Depending on which "flag" (*standard_product_id_type*) - EAN or ASIN - down in the inventory management, the respective value is passed to the Speed4Trade CONNECT.

8.2.1.4 Item -> Amazon -> Category Data

Country-specific data set, for example, for Amazon.de.

The so-called "*RecommendedBrowseNotes*" can be either for each article individually for each product group or set.

Speed4Trade CONNECT field name	Description	Field name Webservice
RecommendedBrowseNotes 1	Specifies the proposed search node 1 of the article.	recommended_browse_node1
RecommendedBrowseNotes 2	Specifies the proposed search node 2 of the article.	recommended_browse_node2

Item -> Amazon -> Prices

The price of an item can be transferred for each sale channel account, which is stored in the Speed4Trade CONNECT (account-specific data). A better alternative is the transfer of the standard price via the pricelist feature centrally.

Speed4Trade CONNECT field name	Description	Field name Webservice
Standard price	Standard price for Amazon	price
Special / Start date	Start Date for Special	sale_start_date
Special / end date	End Date for Special	sale_end_date
Special offer / Price	Price of package	sale_price

Item -> Amazon -> Settings & quantity

<u>Note about Fulfillment by Amazon:</u>
The value "AFN" means that the item ordered will be shipped via an FBA warehouse. The value "MFN" is set, then the article over another / own camp will be sent to buyers.

Currently, about Speed4Trade CONNECT the same item under one account only once be set at Amazon.

Speed4Trade CONNECT field name	Description	Field name Webservice
Start date	Here about the date can be defined date when the product is available.	launch_date
Lead time of delivery	Is taken from the article Basic Data (delivery time)	fulfillment_latency
Switchfulfillment to	- the value "AFN" must be set if an inventory on the FBA warehouse is available - the value of "MFN" must be set if no stock on the FBA warehouse is gone and needs to be sent the article on to another camp.	switch_fulfillment_to
State	State of an article	condition_type
Condition	More detailed description of the condition	condition_type_note

8.3 Marketplace eBay

Speed4Trade CONNECT path: Item → eBay → Category

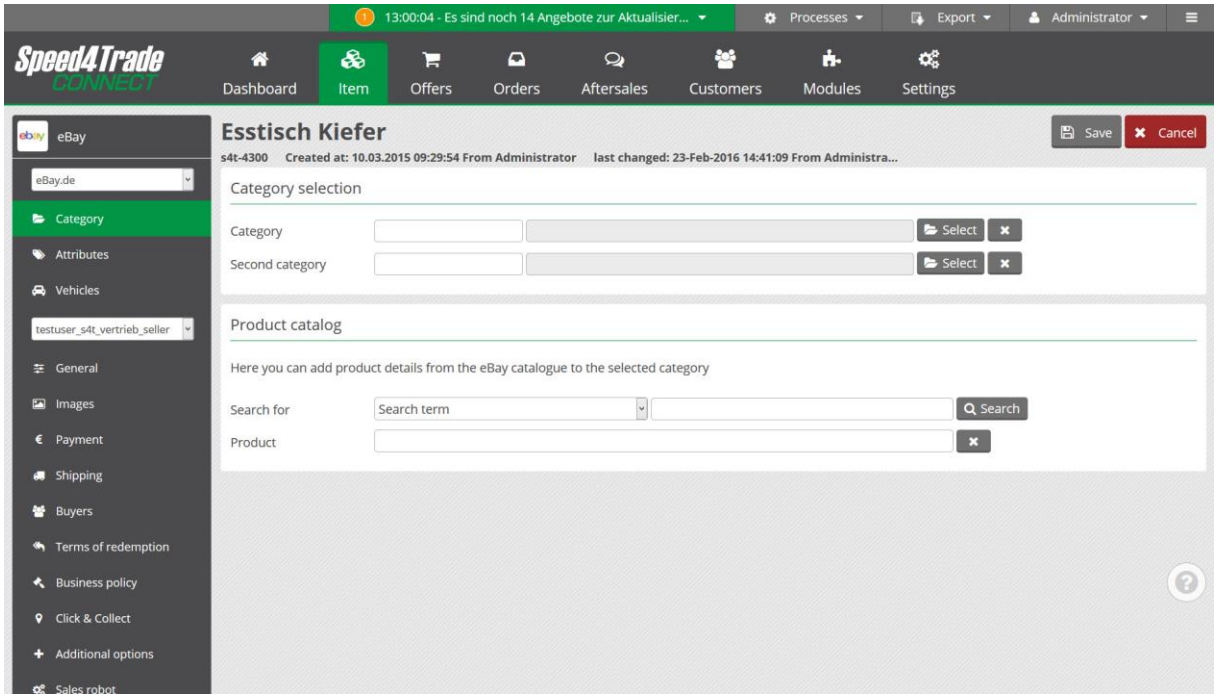

Illustration 26: eBay Category

Similar to Amazon, there is also eBay settings, apply the account independently for the product depending eBay platform. For this example counts the eBay category.

All other settings can be set individually for each account, such as the eBay title or the price.

Required preparatory Calls:

- *fetchSubChannels* (Sub-Channels)
- *fetchAccounts* (Accounts)

8.3.1 Webservice Call: updateItems (eBay-specific data)

Item -> eBay -> Categories

Country-specific data set, for example, for eBay.de.

Speed4Trade CONNECT field name	Description	Field name Webservice
Category	Required field	ebay_category_id_1
Second Category	Attention! Causing duplicated listing fees on eBay, if set	ebay_category_id_2

When are deposited by different eBay countries in the Speed4Trade CONNECT eBay account, then you have each eBay country also has a category set (in Speed4Trade CONNECT, via Webservice).

Should be shown otherwise in the ERP of the eBay category tree, Speed4Trade can provide a file containing the data.

Item -> eBay -> Attributes

Set country-specific data, for example for eBay.de.

Speed4Trade CONNECT field name	Description	Field name Webservice
Artikelzustand	Pflichtfeld	item_condition

Depending on which was chosen eBay category or deposited, category - specific attributes are displayed. This can also be set via the commodity economy. However, it is not recommended in a simple integration model of it , because the attributes are very dynamic and must be taken to ensure that the values have to be written as well as on eBay. If the attributes are written differently, then be created new. However, these are not considered by eBay in the search. For this reason, the maintenance of the eBay attributes in the rule in the Speed4Trade CONNECT done.

Item -> eBay -> General

The following settings are stored in the respective eBay account and passed on to the relevant article. These must not be set by the commodity economy.

- faceplates
- Payment Options
- Free Shipping (categories)
- buyers circle
- Additional Options

Required preparatory Calls:

- *fetchAccountsCategories* (categories)

- *fetchTemplates* (listing templates)

Speed4Trade CONNECT field name	Description	Field name Webservice
Item description	Required field (maximum. 80 sign)	title
Subtitle	Attention! Paid, may, inter alia, also be used for basic price information.	sub_title
Shop category	eBay Shop category that can define yourself any eBay sellers. Set via WebService or manually in Speed4Trade CONNECT the item or multiple editing. Alternative: Care about goods group → Inheritance	shop_category_id_1 (*fetchAccountCategories*)
Second Shop category	eBay Shop category that can define yourself any eBay sellers. Set via WebService or manually in Speed4Trade CONNECT the item or multiple editing. Alternative: Care about goods group → Inheritance	shop_category_id_2 (*fetchAccountCategories*)
Start price	Start price	minimal_bid
Buy it now price	Buy it now price	buy_it_now_price[2]
Best offer	Accept proposed price generally	best_offer_enabled
Accept best offer	Suggest accept from	best_offer_accept_threshold
Refuse best offer	Proposed price decline under	best_offer_reject_threshold
Proposal template for highest bid	Assigning a listing template. Can be done via multiple editing in Speed4Trade CONNECT , or one gets the offer templates in advance via fetchTemplates from. Alternative: Care about goods group → Inheritance	auction_template_id
Proposal template for a fixed price	Assigning a listing template. Can be done via multiple editing in Speed4Trade CONNECT , or one gets the offer templates in advance via fetchTemplates from. Alternative: Care about goods group → Inheritance	template_id_bin

[2] Prefered implementation via price lists instead of this channel specific price!

9 Overview of marketplace-specific characteristics

	eBay	Amazon	Yatego	Rakuten
URL-images	✓	✓	✓	✓
local-images	✓	-	-	-
Length of the Description	80 characters	200 characters *[3]	255 characters	100 characters
Length of the item description	65536	2000 characters	*[4]	No restriction
HTML text	No restriction	Bold, italic, underline, enumeration	Bold, italic, underline, enumeration	Bold, italic, underline, enumeration
Plain text	✓	✓	✓	✓
Returns management	in Speed4Trade CONNECT	On marketplace	On Marketplace	On Marketplace
Template	✓	✓	✓	✓
Price	Fixed price Starting price	Standard price Special price	Gross price UVP	List price Special price
Shop	✓	Only for dealers Platinum Store authorization	✓	✓
Representation of variants	✓	✓	✓	✓

[3] reduced by Amazon from 500 to 200 effective 15.7.2015
[4] The number of characters for the Yatego short description is 130 characters. In Long Description there is no character limit.

10 Inventory Management

Inventories must be regularly passed from the ERP system and as current as possible at the Speed4Trade CONNECT. In the article management, there is a separate area for inventory management.

Speed4Trade CONNECT path: Item → Basic Data → Stock management

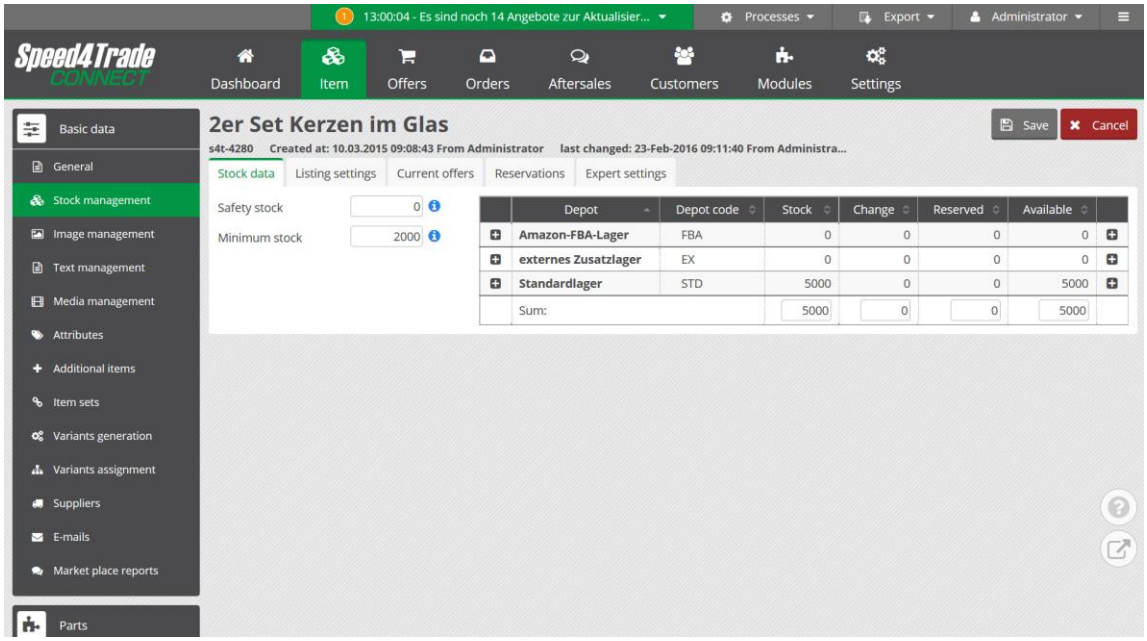

Illustration 27: Stock management

There are 3 important component to fields that must be set:

- Stock (stock_count)
- safety stock (safety_stock_count)
- Minimum Holding (minimal_stock-count)

And yet two optional quantity fields for setting the range that can be set:

- setting amount (placing_count)
- replenishment (refill_count)

The Speed4Trade CONNECT in this case has a so-called "interactive inventory adjustment" .

<u>This means in detail:</u>
In the "Listing Settings" in the inventory management is used to define a primary sale account. Furthermore, we define each account, with any quantity set of products at the retail channel and at what amount of stock to be replenished.

Example:
Setting amount 10 , replenishing 8: Sinks of stock in the distribution channel under 8 , then the amount will be increased again to 10 . This system is particularly relevant on eBay, to form the so-called "Sales Rank".

Decreases the actual stock under the safety stock, all offers will be completed by the primary sales account. Decreases the stock even under the minimum stock, then the offers of the primary sales accounts will be terminated.

Through the interactive inventory adjustment, it is also possible to sell a stock evenly over different accounts.

Example:
Stock: 5, safety stock : 0 , minimum inventory : 0
The article is set in both Amazon and eBay and the online store with the amount 5.
If, for example an item sold on eBay, then the amount in Amazon and the online store is reduced to 4.

Note:
If -1 is passed for the setting amount, then the interactive inventory adjustment for this offer is disabled.

10.1 stock transfer by item master data

Basically, there are two variants to put stocks in the Speed4Trade CONNECT on the Webservice. One possibility is that the stocks are placed on the Webservice call "update items", which is also used for setting the item master data. The second possibility is that the stock on the Webservice call "update stock count" is set. We always recommend the stock on the Webservice call to put "update stock count", as this is much more efficient.

Notes on the stock transfer :

1. The stock transfer should - as mentioned - regularly and as promptly as possible to the event of inventory change in the external system done, that the Speed4Trade CONNECT can adjust the stocks in the sales channels , if necessary , so that , for example, occur no sales.
2. The "Listing Settings" can also be set via the Webservice. However, it is also possible that one puts this on a "Mehrfacheditier"-function in Speed4Trade CONNECT.
3. Setting and "Auffüllmengen" can be preset using the sales account configuration. This default is assumed, however, initially only with creating a new article.

10.1.1 Webservice Call : update items (itemset)

Standard

When using "update items" do not necessarily getting all the fields are passed, i.e. when setting the item master data, you pass all the possible values from the external system, when setting the herd is the appropriate value from the external system. So you can work for the stock update with a relatively "lean" call.

Possible events for the stock transfer:

- Goods receipt
- Telephone / manual jobs in the external system
- Returns / returns
- ...

Required preparatory Call:

- *fetchWarehouses* (stock)

10.1.2 Item -> Basic Data -> inventory management

Speed4Trade CONNECT field name	Description	Field name Webservice
Stock	Camp is to be set for the inventory	Warehouse_id (Abfrage über *fetchWarehouses*)
Current stock	Absolute amount of stock to the article	stock_count
Safety stock	Value is to be terminated as of the offers from the market places, except for primary sales account	safety_ stock_count
Minimum holding	Value, as of the deals to be completed at the market places, also applies to primary sales account	minimal_ stock_count
Setting amount	Amount should be listed with the items on a marketplace	placing_count
Replenishment	Amount is to be held from the refilling	refill_count
Primary sales account	Can be set only once per articel	primary

10.2 stock transfer by separate stock update

Required preparatory Call:

- *fetchWarehouses* (stock)

10.2.1 Webservice-Calls: updateStockCount (Artikelbestände)

Standard

Speed4Trade CONNECT field name	Description	Field name Webservice
Item description	Required field	identifier
Internal Item number	Required field	internal_item_number
Current stock	Absolute amount of stock to the article.	stock_count
Stock	Camp is to be set for the inventory	Warehouse_id (Abfrage über *fetchWarehouses*)

11 Order Management

The functions of order processing such as cash balance, invoicing, reminders usually take place centrally in retail management. About the Speed4Trade CONNECT the orders are regularly automatically picked up from the market places. These are then directly to the ERP for pickup by Speed4Trade-CONNECT-API-Webservice interface. Thus, the orders in the ERP system can be differentiated according to, a separate order number range should be used in Speed4Trade CONNECT. The marketplace eBay may for example get the prefix "EBY", Amazon, for example, "Ama" and the shop, the prefix "Shp". On the basis of these distinctions appropriate analyzes can be performed in retail management. Below are two possible order workflows are explained in more detail.

11.1 Order Workflow without an external system

Within the Speed4Trade CONNECT there is a set procedure for the sale of goods, which is also reflected in the main navigation.

Illustration 28: Navigation bar

The starting point of all are products. From the articles "Prepared deals" are. After uploading to the sales channels they become "ongoing opportunities'. If there is no buyer at an offer or is an ongoing offer complete, then you will find this under "Completed listings".

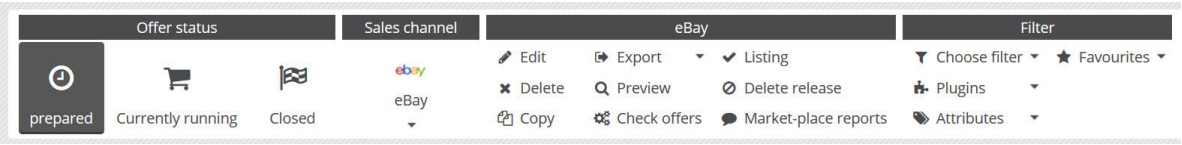

Illustration 29: offer workflow

In the case of a sale the contract is automatically picked up from the respective sales channel and there is a new entry under "orders".

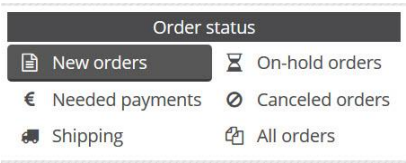

Illustration 30: order workflow

For each order in "New orders" an Initial Notification is sent. Subsequently, the order in the default case will be moved to the status of "Open payments". After receipt of payment (CSV, PayPal) the order is passed to the status of "shipping". From here, the export to shipping services, printing an invoice and in particular the completion of the job are possible.

11.2 Order Workflow with an external system (ERP / MRP)

The workflow to the application system remains the same even when connected to an external system. Depending on the scope of the external system, there are different scenarios when an order is processed by the external system:

1. Speed4Trade CONNECT fetches only start printing. All other processes in the external system.
2. Speed4Trade CONNECT picks up the order and sent the Initial Notification. All other processes in the external system.
3. Speed4Trade CONNECT gets the job after it sends the Initial Notification and performs a comparison of the number of inputs by. All other processes in the external system.
4. Speed4Trade CONNECT picks up the order, sent the Initial Notification, performs a comparison of the number of inputs, creates exports for shipping service and the invoice. The order will be picked up only to statistics from the external system.

Apart from the general order data, invoice and delivery address and sale channel specific position data is returned in the positions, such as the eBay item number.

Speed4Trade CONNECT path: Order → order data → Basic Data

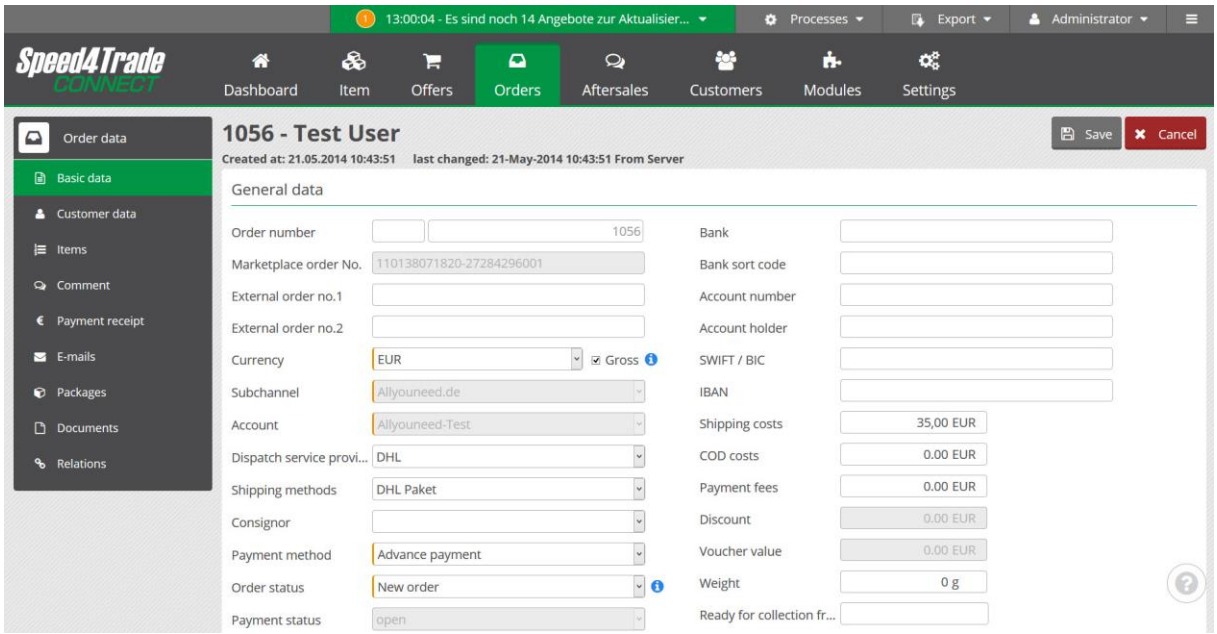

Illustration 31: order data

11.2.1 Default Job Workflow

The definition of the workflow should be discussed in each case with the Speed4Trade GmbH, so that there are as few "breaks" or twice the work for the user. Below an example of a possible job workflow:

1. customer purchases to Buy / Online Store
2. Pick up the orders by Speed4Trade CONNECT
3. Shipping the initial notification-email to buyer by Speed4Trade CONNECT
4. Pick up the orders from the ERP system via interface -> Webservice Call: fetchOrders
5. Setting the "ERP export" flag (the field export_date contains a date) and the external order and customer number -> Web Service call : update orders
6. implementation of the payment balance in retail management
7. status message to Speed4Trade CONNECT -> Web Service call : update orders
8. Automatic sending of the payment receipt e- mail to the buyer Speed4Trade CONNECT
9. transfer of status to Buy / Online Store by Speed4Trade CONNECT
10. Execution of sending in retail management
11. status message to Speed4Trade CONNECT -> Web Service call : update orders
12. Automatic dispatch of the shipping confirmation e- mail to the buyer Speed4Trade CONNECT
13. over of the status Marketplace / Online Store by Speed4Trade CONNECT

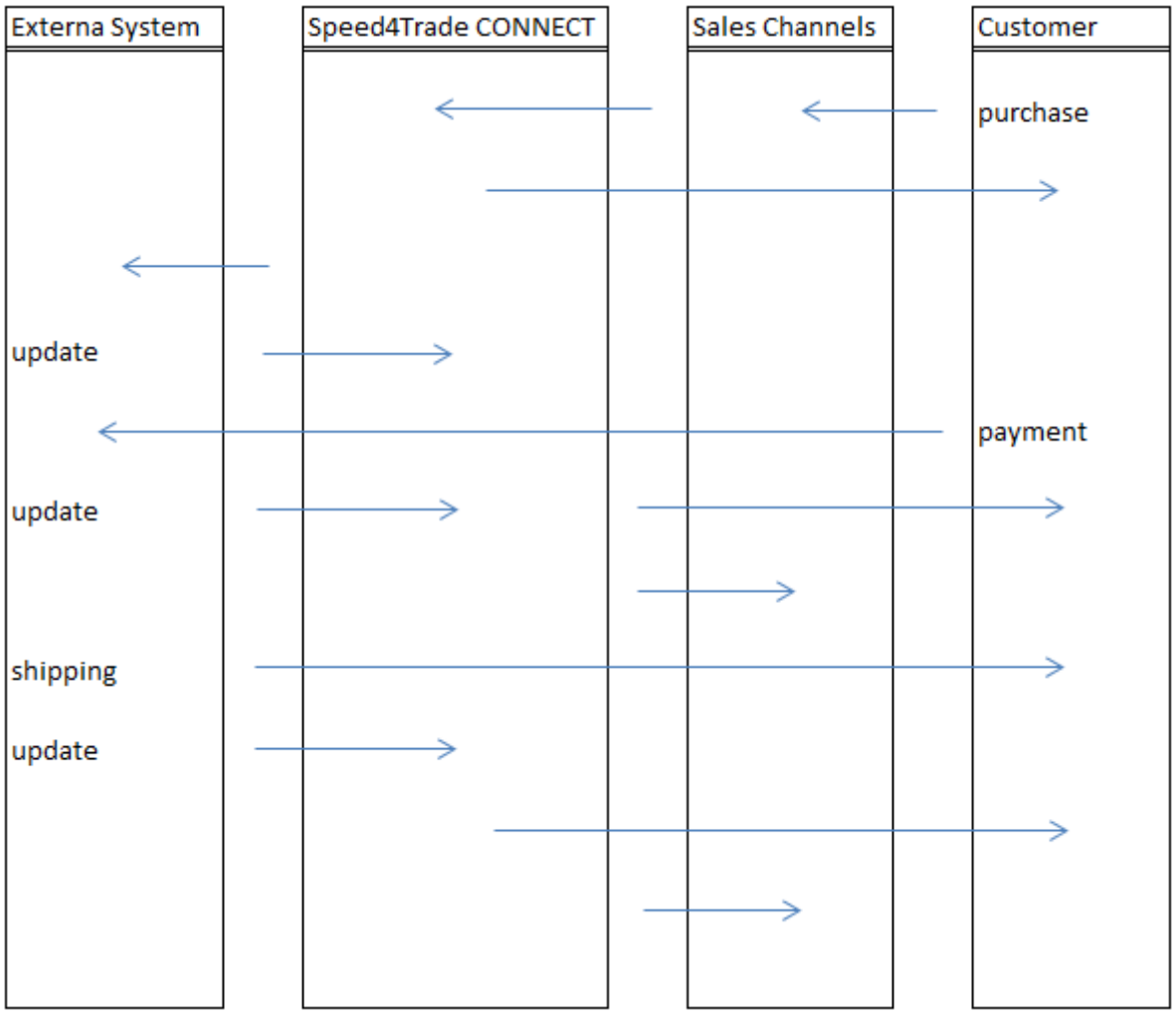

Illustration 32: order workflow

Note!
The order number from the external system should be stored in the field " External order number 1", that is, after the collection of the orders should be equal to an update of the job to be performed.

As an inventory change is made in the external system through the Order Import , also equal to an inventory should be updated.

11.2.1.1 Duplicate Address Check

The orders are always provided with the order and customer data in Speed4Trade CONNECT. In the Speed4Trade CONNECT the duplicate check of customer data on the e -mail address is.

If the ERP have an extended duplicate check, this is to be connected. It may happen that buyers

purchase through the different sales channels with different email addresses. In any case, the buyers at Amazon is assigned its own, generated by Amazon, e-mail address.

11.2.1.2 orders with customer comments

Orders that include a commentary by a buyer should be treated separately in the inventory management, because they must be approved by an employee of the seller.

11.2.2 Order workflow for eBay orders without a completed eBay Checkout

1. customer purchases on eBay marketplace and does not complete Checkout
2. Pick up the orders by Speed4Trade CONNECT with my default
3. shipment of Initial Notification e- mail to Buyer Speed4Trade CONNECT
4. Pick up the orders from the ERP system via interface -> Webservice Call: fetchOrders
5. Creating example a so-called eBay "Vorbelegs" (information document) in retail management to inventory reservation
6. customer completes checkout process
7. picking up the "Complete " status from the inventory management via interface -> Webservice Call: fetchOrders
8. Setting the " ERPF export " flag and the external order and customer number -> Web Service call : update orders
9. implementation of the payment adjustment in retail management
10. status message to Speed4Trade CONNECT ◊ Web Service call : update orders
11. Automatic sending of the payment receipt e- mail to Buyer Speed4Trade CONNECT (or communication automatisms of ERP use)
12. over of the status Marketplace / Online Store
13. for mailing in retail management
14. status message to Speed4Trade CONNECT ◊ Web Service call : update orders
15. Automatic dispatch of the shipping confirmation e- mail to Buyer Speed4Trade CONNECT (or communication automatisms of ERP use)
16. over of the status Marketplace / Online Store

If an eBay contract picked up by Speed4Trade CONNECT, in which the checkout is not completed, then a so-called eBay "Vorbeleg" should be invested in the commodity economy. This is to ensure that in retail management can take place already an inventory reservation. To recognize these orders on "Complete" status. Once the checkout has been completed on eBay, still needs to be performed once the Webservice Call "fetchOrders". Then (contains a date field export_date) is the Webservice call "update orders" the flag "ERP Export" set. Following this, the order can be processed according to the goods economy.

Note!
A more open checkout on eBay could be completed by parameter setting by the Speed4Trade CONNECT after a defined period. But this is not very customer friendly.

But this is not very customer friendly.

Better - depending on the inventory management options: For example, if is an eBay "Vorbeleg" available to a job in retail management and it can be allocated to this one payment, then the "Vorbeleg" to be automatically converted into an invoice. After the Webservice call "update orders" must be carried out so that the "ERP export" flag, external order and customer number is set in the Speed4Trade CONNECT. Speed4Trade CONNECT always closes when setting the "ERP export" flags the checkout from eBay.

11.2.3 Order workflow for Amazon orders - waiting for e- mail

Since there is no direct checkout status at Amazon, must also be checked on the order status here.

The workflow for Amazon orders can be made as follows:

1. Order is picked up by the header data (no e- mail address) =
   Job Status " backlog " (ID 995) -> Speed4Trade CONNECT rider - Residues
2. Order is confirmed as a valid process of Amazon (with email address) = job status "Confirmed / Send" (ID 220) -> Speed4Trade CONNECT Reiter - dispatch
3. Order will be shipped as checked (manually or by ERP) = job status "shipped / Waiting for payment " (ID 230) -> Speed4Trade CONNECT Rider - Open Payments
4. Order is marked as paid by Amazon = job status " Paid / Completed " (ID 240) -> tab Speed4Trade CONNECT - All orders

Amazon orders may be picked up , therefore, only when the job status "Confirmed / Send" (ID 220) is set. In the web service call for Amazon orders must consequently also be restricted on the status of 220 at <order_status_id> next to the flag <export_date>.

Note!

re 3. Sending of the Amazon „Shipping-Status“
The „shipped“ Status (ID 230) may only be send on bulk and not individually for each order.

re 4. Sending of the Amazon „Payment-Status“.

Option 1: „Best Practice“
Speed4Trade CONNECT picks up the Amazon Settlement-Report and processes it automatically.
The external system may not send an “payed” status to Speed4Trade CONNECT.

Option 2: „controlled by external system“
The external system delivers the „payed status“.
The Settlement-Report-Pickup in Speed4Trade CONNECT must be deactivated!
The external system may only update the „payed status“ for orders with an Status (ID 230) in Speed4Trade CONNECT.

11.3 Webservice-Call: fetchOrders

Standard

With the Webservice Call fetchOrders all orders from the Speed4Trade CONNECT can be retrieved. Note the extensive filtering options for Call (orders from / to, only with certain job status, etc.)
Required preparing Calls:

- fetchMandators (client query)
- fetchSubChannels (Sub - Channels query)
- fetchAccounts (Accounts query)

11.3.1 Relevant fields in order

Below is a list of the most commonly used fields when retrieving orders. A complete list is described in the Speed4Trade-CONNECT-Webservice-API online documentation.

11.3.1.1 Orders -> Basic Data

Speed4Trade CONNECT field name	Description	Field name Webservice
Order number	Prefix + Speed4Trade CONNECT order number (Separate fields!)	order_number_prefix order_number
Marketplace order number	Will be delivered by the market places such as eBay, Amazon, and from the shop. Must be processed by the ERP	marketplace_order_id
External order number 1	Can be set by the inventory management, so that you can search in both systems using the same number.	external_order_number_1
External order number 2	Can be set by the inventory management, so that you can search in both systems using the same number.	external_order_number_2
Currency	Currency of the contract	currency_code_iso
Gross	net / gross 0 / 1	price_type_identifier, alternativ: price_type_id
Subchannel	Is transferred from Marketplace	sub_channel_identifier
Sales account	Is transferred from Marketplace	account_identifier
Shipping service		shipping_provider_name
Shipping	Is transferred to the part of market place with.	identifier (im wrapper element dispatch_type)
Mode of payment	Is transferred from Marketplace	payment_method_identifier
Order status	Is set in the workflow of the inventory management	order_status_identifier
Status of Payment	Is set on the ERP	payment_status
Provision reasons	ID of a default reason for the temporary suspension of application workflows.	hold_reason_id

Provision Comments	"Fulfillment By Amazon" → mark of a FBA order from Amazon	hold_reason
ERP Export	Is set from ERP Web service call: update order	export_date
Bank	If necessary, set only from a Store	bank_name
bank code number	If necessary, set only from a Store	bank_code
account number	If necessary, set only from a Store	bank_account_number
account holder	If necessary, set only from a Store	depositor
SWIFT / IBC	If necessary, set only from a Store	swift_code
IBAN	If necessary, set only from a Store	iban_code
Returns	Is transferred from Marketplace	shipping_charge
delivery costs	Is transferred from Marketplace	cash_on_delivery_charge
Payment and fees	If transferred from the marketplace, depending on the payment provider used there	payment_method_charge
Discount	Can both positively and negatively be	rebate
Voucher value	Can both positively and negatively be	coupon_percent
Weight	Weight in grams	weight_in_gram
Order Flag	Priority Shipping und Click&Collect	Wrapper Element order_tags

11.3.1.2 Orders -> packages

Speed4Trade CONNECT field name	Description	Field name Webservice
Shipping service	must be informed of ERP	Call *updateOrders* -> shipping_carrier_identifier
packet number	must be informed of ERP	Call *updateOrders* -> tracking_number

If the buyer a tracking code will be transmitted for the order tracking (by e -mail or, for example eBay, Amazon) then have to be set by the commodity economy, the fields listed above in the transfer of status "Shipping". It can be passed multiple packets of a job with different tracking numbers to Speed4Trade CONNECT.

11.3.1.3 Job Status : regrouped jobs on eBay

In an eBay -commissioned, it may happen that this has the order status "regrouped" . This status is displayed when a buyer brings together two jobs into a single job, while the respective Checkout was not yet complete. The orders, which have the status "regrouped" have no more order items.

<u>When processing regrouped eBay orders please note the following :</u>
If a buyer buys each an item on eBay , so that two jobs are created and the checkout does not terminate , then these two orders of the Speed4Trade CONNECT be picked up with the defined default payment method. Subsequently, those economic fetches the jobs from the Speed4Trade CONNECT via the interface for further processing. In the ERP are specified on the inventory posting two information documents - one per order - created. Is to orders already a receipt of payment available , then the Info

documents are automatically converted to invoices in the ERP system . If now the parallel jobs on eBay grouped by the buyer to one, then the jobs are summarized in this Speed4Trade CONNECT to one when re- fetching . This means that the positions are associated with a job . Thus, the rearranged orders can be picked up from the inventory management through the interface of the Speed4Trade CONNECT, must be called with an additional filter criterion again the Webservice Call " fetchOrders " . Below you will find an example of this call.

```
<request method="fetchOrders" version="1.0.0">
<filter>
      <mandator_id filter_method="=" filter_value="1" />
      <order_status_id filter_method="=" filter_value="985" />
      <last_changed from_date="2012-01-01T00:00:00.000+02:00"
allow_null="true" />
</filter>
</request>
```

11.4 Webservice Call : update orders

Standard

The update of the order in Speed4Trade CONNECT by the external system is a very important process to inform the buyer about the current status of his order , and thus to reduce communication overhead with the buyer.

Example scenario :

1. Only Initial Notification in Speed4Trade CONNECT
2. Rest of the process in the external system
3. This is an eBay order with payment method "cash in advance "

After receipt of payment in the external system of the "paid" status to Speed4Trade CONNECT is fitting. Speed4Trade CONNECT in turn passes this status on eBay, so that the buyer can recognize this in his "My eBay " section .

After shipping the goods the external system are these "shipped " status - in the best case, including the tracking number - to Speed4Trade CONNECT on. Speed4Trade CONNECT then sets the status back to eBay and transfers - if any - is the tracking number.

Especially with Amazon, it is very important that the "shipped " status is passed . Every order at Amazon must be marked "shipped" as mandatory in order to obtain the money from Amazon for this order. This is not done within a certain period , then goes Amazon believe that the goods were not shipped and the money will be remitted back to the buyer.

Note:
Creating a new order by this call is only possible when it comes to the channel "Manual Sales", for example, Telesales. Orders to marketplaces or shop systems may not be applied in this way, because it can lead to errors in coordination with the marketplaces or shops in the subsequent process.

Required preparatory Calls:

- fetchMandators (clients)
- fetchSubChannels (sub - channels)
- fetchAccounts (accounts)

12 Additional Functions

12.1 Initial Notification-Mail to buyers

The notification to the customer of the order status is done via e-mail about the Speed4Trade CONNECT or ERP. This must be determined in each specific case. In no event e-mail should be sent twice by both systems. It is possible in Speed4Trade CONNECT, set the automatic shipping for each status.

12.2 Payment balance

The payment adjustment takes place directly into the inventory management is usually through PayPal and bank account. The settlement reports from Amazon can be read directly from the Amazon Web service at Speed4Trade CONNECT and can be picked up in the ERP system via " fetchPayments " .

When setting the pay status of the interface , the entire gross amount of Speed4Trade CONNECT is always transmitted.

Note:
At Amazon the entire fetchReports unprocessed Settlement Report can be collected, which contains more detailed information.

12.2.1 Webservice-Call: fetchPayments

If your system have no way of payment data to match (eg PayPal), this can also be done via Speed4Trade CONNECT.
The allocation of payments for orders is thus significantly facilitates the external system.

The workflow is as follows :

1. The orders are collected by the external system from Speed4Trade CONNECT.
2. Speed4Trade CONNECT picks up on the automation of PayPal payments .
3. About the fetchPayments call all numerical inputs can be determined. Number of inputs that could be assigned automatically by Speed4Trade CONNECT, while the same contain the reference to the order.

12.3 Delivery of orders via FBA

The orders may be sent via FBA only if the entire order can be processed on top, so all positions with all quantities can be delivered. The shipping of Partial Reels or individual items of an order for FBA is not possible.

Note!
For example, orders can be sent from the own online store via FBA , appropriate for this web service calls must be performed. Since this extension of the web services is still in the realization of the corresponding call to be completed at a later point in this document.

12.4 Partial deliveries

It can be handled via Speed4Trade CONNECT no partial deliveries. Should it come in retail management is a partial delivery , then it must be decided in the connection , whether the contract in Speed4Trade CONNECT after the first partial delivery or only after the complete delivery of the goods is set to " completed / sent " set .

12.5 After Sales

12.5.1 eBay

The delivery of reviews and the triggering of inconsistencies from the after-sales area can be triggered via Webservice calls by the commodity economy.

12.5.2 Amazon

Adjustments and cancelations of Amazon orders will be executed in two steps:

1. Execute the order cancellation in Speed4Trade CONNECT
2. Execute the order cancellation at Amazon

Full order cancellation:
- Use call **updateOrders** to set the order_status_id to 990 (canceled).
- Execute the call **cancelOrders**

Partial order cancellation:
- Use call **updateOrders** to adjust the quantity of the relevant order_item. Do not adjust the order_status_id.
- Execute the call **adjustOrders**

Refund (no return of ordered items by buyer):
- No adjustment of the order required.
- Just execute the call **createCreditEntry**.

13 Remote control of Speed4Trade CONNECT

13.1 General

If an item on Speed4Trade CONNECT is passed, then this is not automatically adjusted in a market place or shop. However, if an ERP system can provide all the data necessary for setting, then all relevant processes can be initiated.

13.2 Workflow for automatically adjusting

The procedure is as follows:

5. The Call updateItems an article with the marketplace- specific data in Speed4Trade CONNECT is created. The response to the Speed4Trade CONNECT API internal item_id of the article.
6. Via prepareOffers and the item_id for example, you can create an offer for eBay. This gives back a offer_id.
7. This may offer_id in turn are used in the call upload offers to adjust the item directly on eBay.

Note:
Please note on the site that uploading a bid runs asynchronously to a market , ie Although upload offers abuts the upload process , the actual upload may be slightly offset in time so that the response of upload offers only indicates that the request has been received . But gives no indication whether the upload has run flawlessly for the marketplace itself. But you can determine by fetchOffers if it's still that special "Prepared offer" are after a certain time .

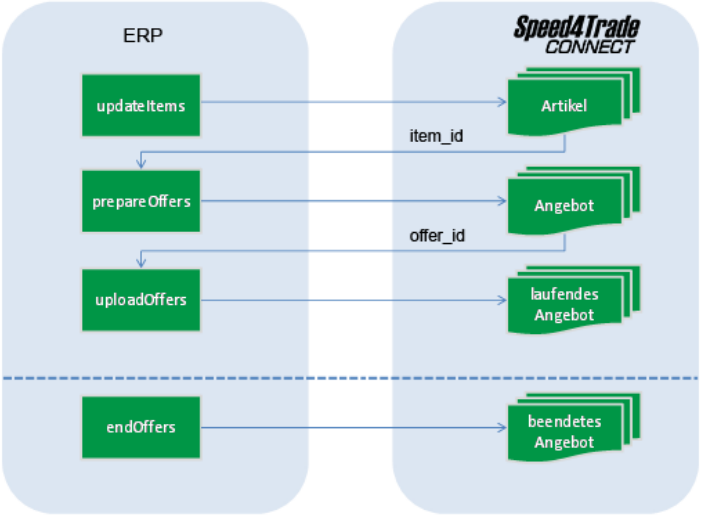

Illustration 33: Workflow for automatically adjusting

13.3 Balance of new product data into marketplaces

Except for the stock new product data are not synchronized automatically with marketplaces. New product data can be updated either via the Speed4Trade CONNECT client or via syncPlatformItems toward market places and shops.

13.4 Show masks of Speed4Trade CONNECT in a container

In certain implementations, it is desirable that masks are displayed from Speed4Trade CONNECT directly in a commodity economy. For this, the ERP system must have a container e.g. an IFrame , etc. a WebControl can offer , to which a URL is passed . This URL must have a specific structure , thus also log on to the Speed4Trade CONNECT can be done automatically ..

13.4.1 Structure of the mask address for the automatic login

The URL for calling a Speed4Trade CONNECT mask consists of two parts. Firstly, from the default URL for calling the Speed4Trade CONNECT WEB, e.g. https://Servername:Portnumber/emmidaweb/de-de/items/common.html. And on the other, the additional parameters that are necessary for automatic registration of the enterprise resource Speed4Trade CONNECT. These are:

- mid = client id
- user = username in the Speed4Trade CONNECT
- nonce = timestamp
- seckey = signature to authenticate

If all parameters are calculated and assembled with the invoked mask address, resulting in a total URL that is constructed as follows :
https://Servername:Portnumber/emmidaweb/de-de/items/common.html?itemId=3747503&mid=1&user=max_mustermann&nonce=201401101345&secKey=dGXKxOVp2dwnsa8Iz2Itu%2b4QL4kObGkzC%2fi15Pt2h2MgcC0%2bisLbgg%3d%3d

13.4.2 Information about authentication

The user has logged in once, he will remain logged in until they log out again. In order to prevent replay attacks, a time stamp is transmitted as a parameter. If the time stamp is outside a 10 -minute interval, so the login fails or if the user is already registered, no page is displayed.

The parameter name "nonce" is used and contains the time stamp in the format year, month, day, hours, minutes (yyyyMMddhhmm) .

Example of the technical implementation:
parameters:

- itemId=3747503
- mid=1
- user=max_mustermann

- nonce=201401101345

All parameters are assembled as a string (without separators):
Then a signature is for this string via the DSA method created and "seckey" as a parameter as follows to the URL string attached:
https://servername:portnumber/emmidaweb/de-de/items/common.html?itemId=3747503&mid=1&user=max_mustermann&nonce=201401101345&secKey=dGXKxOVp2dwnsa8Iz2Itu%2b4QL4kObGkzC%2fi15Pt2h2MgcC0%2bisLbgg%3d%3d

For the generation of the signature, a private key is generated for the client in Speed4Trade CONNECT, which must be adopted in the ERP and stored there safely. The login key is found in the Speed4Trade CONNECT the Tools section - > Basic Settings - > Client -> Other -> External Systems -> Login - Key.

This key is a DSA key pair in PKCS # 8 encoding in DER and for better portability as a string or Base64 - encoded private key.

To make the connection a little easier , there is the Partner Portal , a set of programming examples, including libraries , source code and demo application for the computation of a signature with DAS . For the two programming languages Java and C # , there is in each case a package to download. This data may be used freely for the connection to the Speed4Trade CONNECT.

13.5 Hiding the framework of Speed4Trade CONNECT WEB Templates

So that the mask better Speed4Trade CONNECT in a container, such as IFrame can be integrated, there is the option to hide the frame. This framework is necessary for stand-alone use to switch between articles, quotes and orders back and forth.

For an additional parameter "frameless" must be specified in the URL yet. It can be specified here 2 values: frameless = 1 turns off the frame, frameless = 0 hides it again. The default value is 0 if the parameter is not specified with.

13.6 Definition of a callback address

About an additional parameter in the mask address (URL) you can control the behavior of the displayed Speed4Trade CONNECT mask after saving. For this there are three different cases, which are illustrated using the example of the maintenance of product data .

13.6.1 Parameters callback not set

If the parameter ' callback ' is not set, the input mask is analogous to a direct call to Speed4Trade CONNECT WEB. This means in the above example in section 13.4.1 when the user is finished editing the mask and click Save, or Cancel, the system jumps to a higher-level mask. In our example, the user edits the data of an item and then passes back to the article list.

13.6.2 Parameters callback = no

If the callback parameter with the value "no" is passed and no Cancel button is displayed. When you click on the Save button, the entries will be saved, but the mask remains open afterwards.

Illustrated by our example above, this means that the user always stays on the current screen for editing an article that was called by the commodity economy. The URL looks like this: https://.../items/commin.html?itemId=4711...&**callback=no** & seckey = ...

13.6.3 Parameters callback put on a different mask

With the parameter ' callback ' can also be controlled directly, the flow between different mask calls. Thus one can determine from the inventory management from which subsequent mask should be displayed according to the current callee. Since no parallel ' callback = no ' can be set, the behavior after calling the result mask is as described in section 13.6.1. That is, the way in which the mask is identical to the standard call of Speed4Trade CONNECT WEB via the browser.

<u>The example of this URL may look like this:</u>
https://.../items/common.html?itemId=4711...&**callback=orders/list.html**&seckey=...
This opens the edit form of the article with the ID 4711. If the user is finished editing the input mask, he can save, or cancel. As a result, the mask with the job list is displayed

14 Summary

The most important settings are the basis of this document, you are shown, you must configure in Speed4Trade CONNECT before you can begin the actual connection. This basic configuration is thereby carried out either by the user himself or in cooperation with the support department of Speed4Trade GmbH.

In order to avoid false expectations created by the user and agreed to the scope of functions, you should contact time with the Speed4Trade GmbH. Thus it can be received close attention in training, what the user must subsequently maintain in Speed4Trade CONNECT, how this can be achieved in the fastest possible way and how the order workflow should look like.

The functions described only form the basic structure that should be present from the perspective of Speed4Trade GmbH. Of course there may also be differences, this depends very much on the requirements of the user.

14.1 Workflow overview

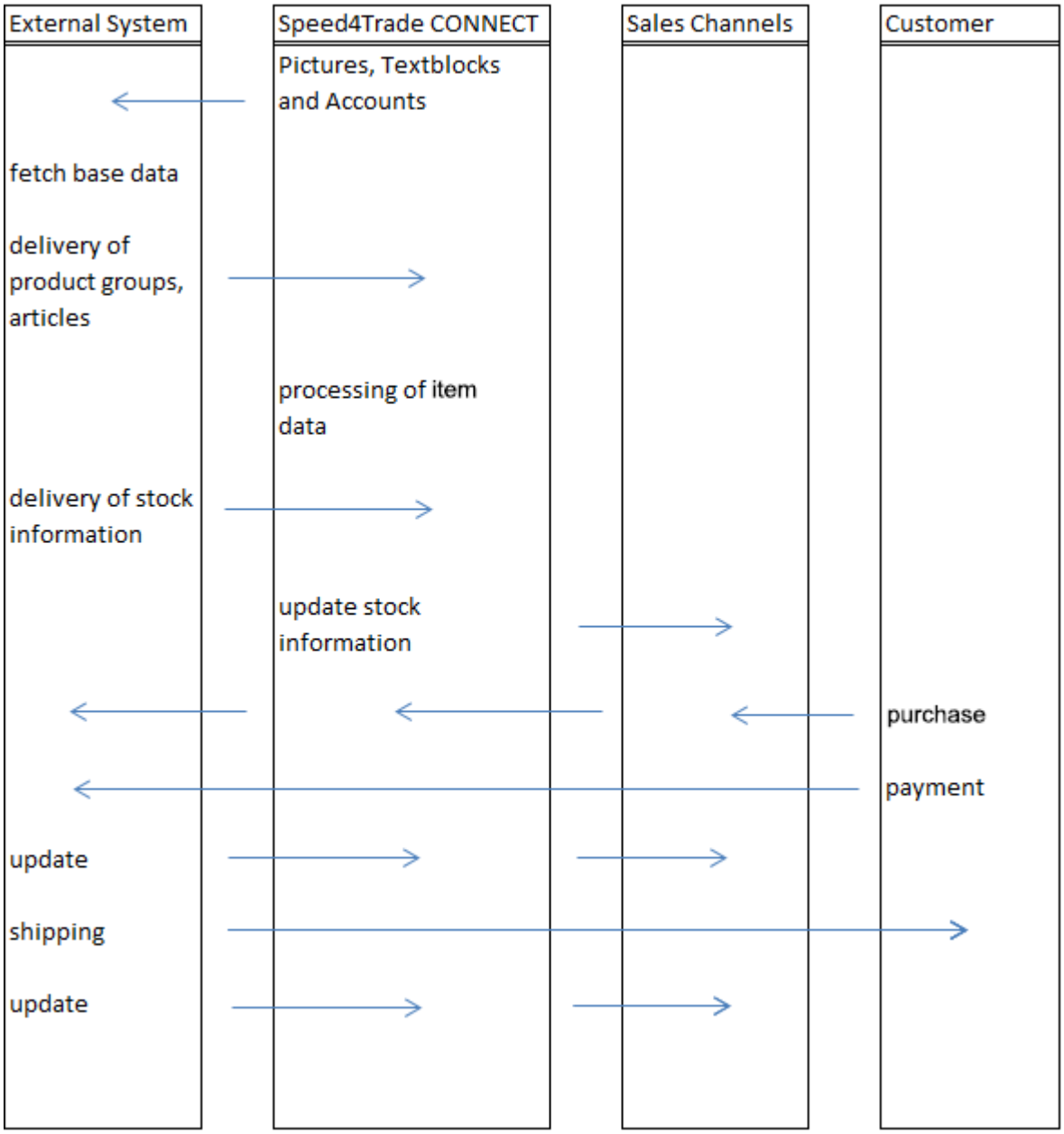

Illustration 34: Overview diagram of the workflow