Implementation Guideline

NMVS Blueprint Core 1.03

Table of content

Tables

Graphics

1 Approvals

By signing/dating in the space provided below, the following representatives indicate approval of the present document.

Arvato approvals

Name:		**Signature:**	
Role:	**(Author)**		
Date:			

Name:		**Signature:**	
Role:	**(Reviewer)**		
Date:			

Name:		**Signature:**	
Role:	**(QA Review)**		
Date:			

Name:		**Signature:**	
Role:	**(Approver)**		
Date:			

2 Change History

Date	Version	Change Description	Author
23.03.2017	0.1	Merge of Documents “Implementation Guideline NMVO 0.91” and “BP 1.0 NMVS API Reference Distributors Interface”	
07.05.2017	1.0	Final	
08.06.2017	1.1	Review and various additions	
07.07.2017	1.2	Various additions	
26.07.2017	1.3	Various additions and Review	
11.08.2017	1.4	Added descriptions regarding example code	
17.08.2017	1.5	Complete Review	
06.09.2017	1.6	Various additions concerning technical overview	
29.09.2017	1.7	Added chapters **Data Matrix Encoding Standard** and **When to Call the NMVS**	
13.10.2017	1.7	Added G101 for BP Core Release 1.1	
25.10.2017	1.8	Updated descriptions regarding example code	
27.10.2017	1.8	Updated permission mapping Added document management	
03.11.2017	1.8	Change in Figure 4	
09.11.2017	1.9	Cleaned up wording	
22.12.2017	1.9	Changed headline and footer	
31.01.2018	2.0	REST Interface	
30.03.2018	2.1	Minor corrections Jira: NMVSBP-642 & NMVSSUPP-208	

Date	Version	Change Description	Author
18.05.2018	3.0	Finalization, adding references, adding new interface version in chapter 8.3	
25.05.2018	3.0	Renaming G611 to G615, replaced wrong XML example	
05.07.2018	4.0	Integration of new graphics and added concentrators to chapter 3	
02.08.2018	5.0	• General review • Added general changes for interface V2 and V3 • Added information on PPN • Adjusted figures on expiry of certificates • Added description on G482 and G483 (T&C) • Added description on limitations on G131, G137, G181 and G187 (undo destroy and undo stolen) • Corrected rights and permissions for bulk transactions • Replaced lists of return codes with references to TD-001	
14.08.2018	5.1	• Added Password Renewal Process (NMVSSUPP-782) • Merged two existing versions • Added web service life cycle description • Removed information to web service V1	
12.10.2018	5.2	• Changes to Permission mapping • Introduction of web service life cycle	
16.11.2018	5.3	• Review & minor changes	
10.01.2019	6.0	• Updates for release 1.03 • Endpoints added	

3 Disclaimer

Copyright

The content of this document are the property of Arvato Systems. No part of this document may be duplicated (in whole or in parts), transmitted to third parties or published without the prior written consent of Arvato Systems.

This document has been created with the greatest possible care according to the state of the art in technology. However, Arvato Systems assumes no guarantee or responsibility for the up-to-dateness and correctness of the information provided.

Concentrators

Depending on the country you develop client software for, the NMVO of your country might decide not to use or allow concentrators to be used. The concept of a concentrator is described here for the sake of completeness, but it doesn't imply that a concentrator is to be used by all means.

Please contact your local NMVO to clarify how to connect to the NMVS and if you are allowed to set up and use a concentrator.

Introduction

The National Medicine Verification System ("NMVS") is the answer to the system requirements based on the EU legislation EU/2011/62 ("Falsified Medicines Directive"). The scope and the specification of the so called 'Blueprint Solution' was defined by the European Medical Verification Organization ("EMVO"). It is made clear that Arvato Systems is solely responsible for the implementation of the specification and **not** for any functional definitions and/or any performance requirements. The specification may be changed or modified from time to time by the EMVO, so that adjustments or changes of the functionality by the publishing date of this document may become necessary. In no circumstances is the respective stakeholder entitled to charge Arvato Systems for any additional costs or work that may arise for the wholesaler / pharmacy software system due to these changes by national authorities.

Arvato Systems will inform the respective stakeholder about changes of the specification, due date of the implementation and the availability of new system features in the NMVS. Software vendors have to consider and plan these changes in their release planning to be compliant with the legislation. Arvato Systems shall be not liable whatsoever for any delays in the wholesaler / pharmacy software system development.

Arvato Systems is also not responsible for the quality of stored product data in the NMVS. Data will be submitted from the Marketing Authorization Holder ("MAH") via the European hub to the NMVS.

4 References

Document-ID	Document Name	Author
TD-035	NMVS - WSDL - XSD_V2.zip	
302006	NMVS – WSDL – XSD_V3.zip	
302003	BP 1.3 ID - Catalog Translations.xlsx (TD-014)	
302004	BP 1.3 ID - G100 Distributor Transactions (TD-001)	
302002	Release Notes NMVS Blueprint Core 1.02+1.03 (ID-021)	

5 Introduction

5.1 Vision & Mission

Falsified / counterfeit medicines are a serious risk for the citizens all around the world. To protect the patients in the EU from the high risk of any impact of this non-valid products, the EU Council and the European Parliament introduced a new legislation that was published on the 9th February 2016 as delegated act. This new legislation is mandatory for all EU members. It is a pan-European strategy to fight illegal medicine products.

To be compliant with this legislation, each pharmaceutical RX product needs to have a harmonized safety feature. This features "should allow the verification of the authenticity and the identification of each individual pack". To make this possible, each pharmaceutical manufacturer has to introduce a unique identifier on every single pack. These pieces of information containing the unique identifier, batch number, product code, expiry date and additional information need to be collected by the manufacturer and to be submitted to a European database. The setup and management of this repository is part of the delegated act as well.

This concept is based on a "Point-of-Dispense Verification" where at least each dispensing point is forced to make a final verification before a handing-over to the patient. This verification procedure is supported by national systems of each member state.

With this concept the EU is reducing the risk of infiltration of falsified medicines into the legal European pharmaceutical supply chain.

The EU legislation has to be locally adopted to become a local legislation in place. The due date for readiness will be 3 years after publishing of the delegated act. This means that all pharmaceutical products in scope will have to have the safety feature, European as well as national repositories will be in place and the dispensing points will have the needed tools in use.

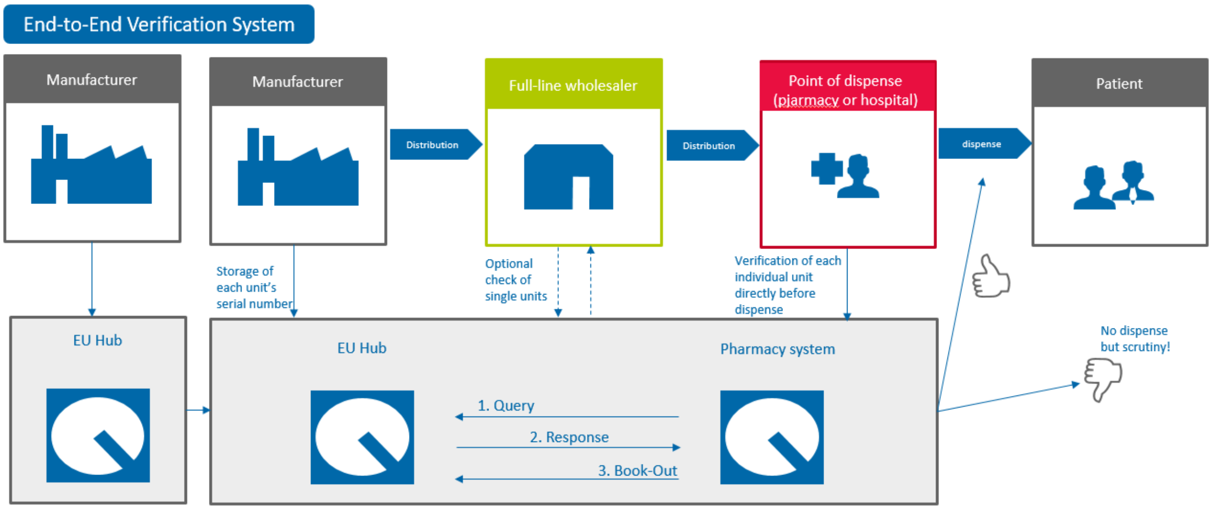

End-to-End Verification System
Manufacturer
Manufacturer
Distribution
Full-line wholesaler
Distribution
Point of dispense (pjarmacy or hospital)
dispense
Patient
Storage of each unit's serial number
Optional check of single units
Verification of each individual unit directly before dispense
No dispense but scrutiny!
EU Hub
EU Hub
Pharmacy system
1. Query
2. Response
3. Book-Out

Figure 1: End-to-End-Verification System

1) © by securepharm

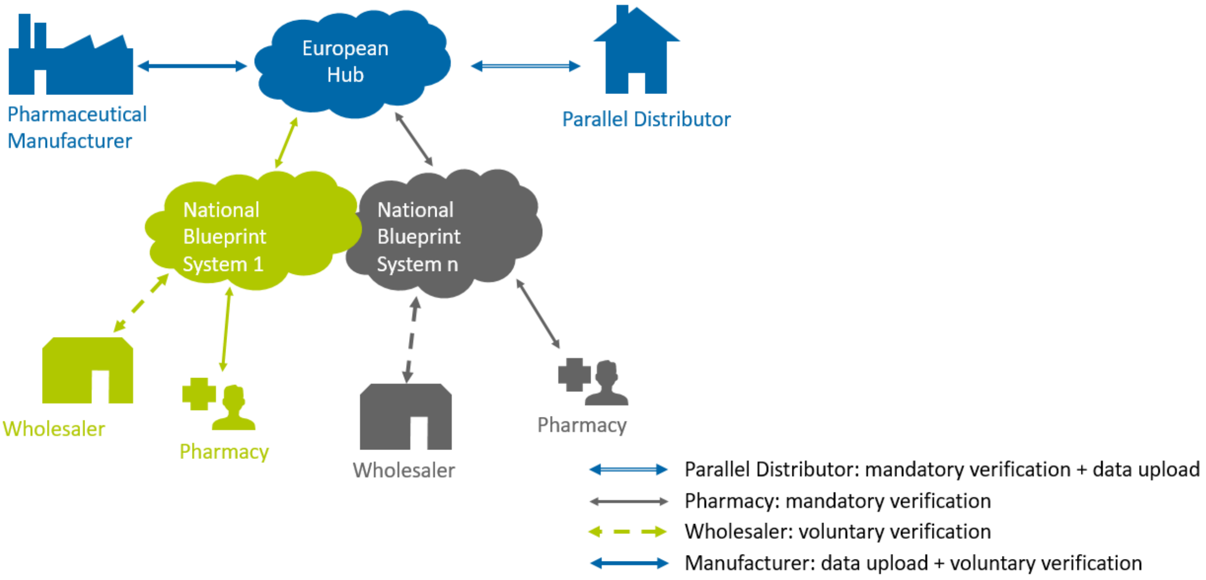

Pharmaceutical Manufacturer
European Hub
Parallel Distributor
National Blueprint System 1
National Blueprint System n
Wholesaler
Pharmacy
Wholesaler
Pharmacy
Parallel Distributor: mandatory verification + data upload
Pharmacy: mandatory verification
Wholesaler: voluntary verification
Manufacturer: data upload + voluntary verification

Figure 2: System overview

5.2 Purpose of Document

This document is an abstract of the European Medicines Verification Organization (EMVO) description.

The document has been written for software suppliers of the stakeholder groups who are authorized or entitled to supply medicinal products to the public and who want to integrate the NMVS API into their software product.

These stakeholder groups are:

- Pharmacies
- Hospital Pharmacies
- Wholesalers
- Dispensing Doctors

The listed stakeholders are also referred to as “Distributors” in the entire document.

This guideline describes the business processes in the National Medicines Verification Systems in a blueprint style. It also provides an overview of the related web services for the distributor interface.

Readers of this document are expected to be familiar with software development, web services and SOAP.

5.3 Reference Documentation and Resources

This guideline is not intended to be a replacement but rather a useful supplement to the “Interface Description” for supporting the integration of the NMVS API, hence the “Interface Description” forms the regulatory basis upon which this document is based.

→ *TD-001 BP 1.3 ID - G100 Distributor Transactions.xlsx*

Further resources serve as technical support for the development project:

→ TD-014 BP 1.3 ID - Catalog Translations.xlsx

Please make sure that all needed prerequisites have been received beforehand. It is recommended to read the rest of this document, to become familiar with terms and concepts.

5.4 Terminology

NMVS	National Medicines Verification System – the system providing this API
NMVO	National Medicines Verification Organization
Distributors	Points of dispense e.g. pharmacies, parallel distributors and wholesalers and more
Client system	The system using the NMVS API
O/M	Optional or Mandatory Field

5.5 Validity of this document

This document might change with a new NMVS release and/or new version of the web services

This document is valid for:

NMVS-Release: 1.03

Web service version: V2 & V3

6 How to start

This chapter will describe how to gather all the relevant information to connect to the national system. Mainly, the way of onboarding is similar for all the distributors (pharmacy, wholesaler and dispensing doctors). Due to this similarity the following chapter describes the onboarding of a pharmacy.

If a distributor develops an IT system to fulfill the EU requirements on his own or has an IT department to develop an own system, the national legislation does not forbid the distributor to have a role as a software vendor for distributors' IT systems besides his origin role as a distributor. Therefore, the distributor needs to consult the separate documentation for software suppliers, too.

Furthermore, there are two different onboarding procedures for the two roles.

6.1 Onboarding of Software Suppliers

The software suppliers are one of the most important stakeholders in the rollout process within a specific country. They need to enable their provided software to interact with the NMVS. To get access to this document, a software supplier has to pass the onboarding process for software suppliers so far and is connected to the Software Supplier Portal.

The Portal can be accessed under the following URL:

https://www.sws-nmvs.eu

We will use the Portal to distribute the latest documentation about the system to you and to inform you about changes in the software.

6.2 Onboarding of a pharmacy

To get all pharmacies in a specific country on board, the NMVO has to provide a list of all known pharmacies in the country. This list needs to contain at least the following pieces of information:

- Name of the Pharmacy
- Postal address
- Contact e-mail address
- Phone number
- Contact person
- User name
- Used pharmacy software and version (optional)
- Pharmacy chain name if applicable (optional)

Based on these pieces of information, the onboarding process of each known pharmacy can start. To minimize problems during the onboarding process, each pharmacy should check if all pieces of information gathered by the NMVO are valid and correct. If there is any mistake in the given data, it could happen that a pharmacy will not get the onboarding documents.

In general, the main coordinator between a distributor and the national system is the NMVO.

6.2.1 Onboarding process description

First of all, the national system will initially create a user account for each submitted user. The previously collected data will be stored in this account. In the next step, the NMVS will create an initial password for that specific account. As a third step the system will generate a specific client certificate and a related passphrase. The user will get the onboarding information for the user account, the initial password and the passphrase in several e-mails or postal.

Within the onboarding information the user will get a link to the certificate portal. In combination with the user account and the initial password, the specific client certificate can be downloaded. The client certificate needs to be stored in the key store of the used browser or the key store of the application. To activate the certificate, the passphrase has to be typed in to open the certificate.

After the activation of the certificate, the registration can be accomplished in two different ways:

Registration via Web GUI:

Together with the registration letter from the NMVO comes a link to the registration page for pharmacies. With the given user data and the respectively submitted initial password the pharmacist is able to start the registration procedure. After a successful registration, the user will be asked to type in a secure password that is only known to the pharmacist and meets the requirements regarding format/range (mix of alpha and numeric character, upper / lower case and at least one non-alpha / non-numeric character) and length (min of 8 digits). The user needs to type in the password twice.

After that procedure, the pharmacist is able to communicate with the national system via Web GUI. The Web GUI functionality will be described further down in this document.

Registration via pharmacy software:

The second way to register is the technical way with the locally installed pharmacy software. In this case, a version of the pharmacy software that is already compliant with NMVS needs to be installed. The pharmacy needs to stay in touch with the software supplier to get the information which software version is ready for connection to the NMVS including the certificate management.

If the right software is installed, the specific function of initial registration will guide the pharmacist through the registration procedure. At the end of the procedure, the pharmacy system is able to connect to the local NMVS.

7 Function Overview / Business Processes

7.1 Overview

The figure below shows the parties involved in the EU legislation EU/2011/62 („Falsified Medicines Directive“). The main focus lays on the interface between the National Repository (National Medicine Verification System - NMVS) and the dispensing points.

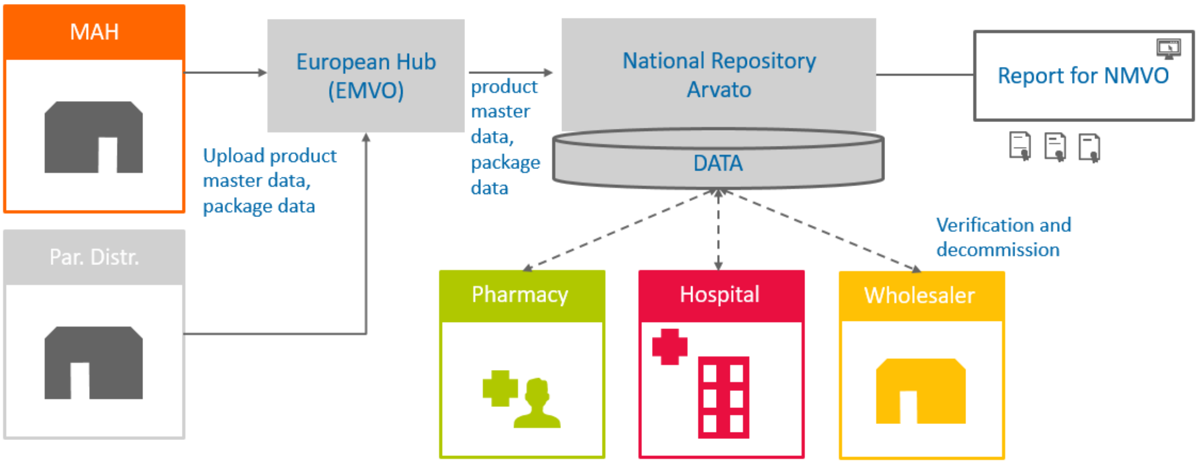

MAH
Par. Distr.
European Hub (EMVO)
Upload product master data, package data
product master data, package data
National Repository Arvato
DATA
Report for NMVO
Verification and decommission
Pharmacy
Hospital
Wholesaler

Figure 3: EU Legislation EU/2011/62 („Falsified Medicines Directive“)

7.2 How to connect to the NMVS

There are different ways for a distributor to connect their software to the NMVS. First of all, there can be the direct connection between a distributor and the national system. Each distributor needs to complete an onboarding procedure to get the user credentials to connect to the system. Each user is known to the national system and can be identified by a user ID and password combination. Access to the national system is possible via the distributor’s IT system as well as via the optional Web GUI (see chapter 13.4.4 G615 - Download Certificate – Request Format)

Attribute	O / M	Description
B – Body data		
B5 - Administration data		
TAN	M	A TAN number to identify a client certificate

Table 1: Body – Download certificate request format

Example:

```
<Body>
  <Tan>126334</Tan>
</Body>
```

Figure 4: Download Certificate – Response Format

Attribute	O / M	Description
B – Body data		
B5 - Administration data		
Certificate	M	Binary data of the private certificate to identify the user.
Certificate name	M	Name of the .p12 file containing the private certificate of user
certificate password	M	Password to access and install the private certificate.
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination Client/user/password invalid

Table 2: Body – Download certificate response format

Example:

```
<Body>
  <Cert name="John002" passphrase="HJ7HK8BKAD0QLIN">
    VBERi0xLjMKJaqrrK0KNCAwIG9iago8PCAvVHlwZSAvSW5mbw...
  </Cert>
  <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
</Body>
```

The second option to connect to the national system is based on a – maybe already introduced – stake-holder member system. For example, this could be a hub system from the pharmacy association with secured connections to the member's systems. In this case, additional pieces of information are required to identify the distributor behind that hub-system by the NMVS. This identification is necessary to enable the distributor to execute all 'undo' related services. The clear requirement is to allow 'undo' services only for the user that accomplished the respective 'do' service. Therefore, a sub-user ID is necessary to identify the end user. The sub-user ID could be the real user as well as an anonymous user. If the sub-user ID is not known either, the only way to perform an ‘undo’ service will be the identification of the respective ‘do’ service by the original transaction-ID.

The system between the NMVS and the endusers concentrates multiple connections of several endusers, say pharmacies, down to a single connection to the NMVS. For that reason the system is referred to as a “concentrator”. Even though it is one logical connection, the concentrator might be linked to the NMVS with multiple physical connections.

Important: Depending on the country you develop client software for, the NMVO of your country might decide not to use or allow concentrators to be used. The concept of a concentrator is described here for the sake of completeness, but it doesn’t imply that a concentrator is to be used by all means.

Please contact your local NMVO to clarify how to connect to the NMVS and if you are allowed to set up and use a concentrator.

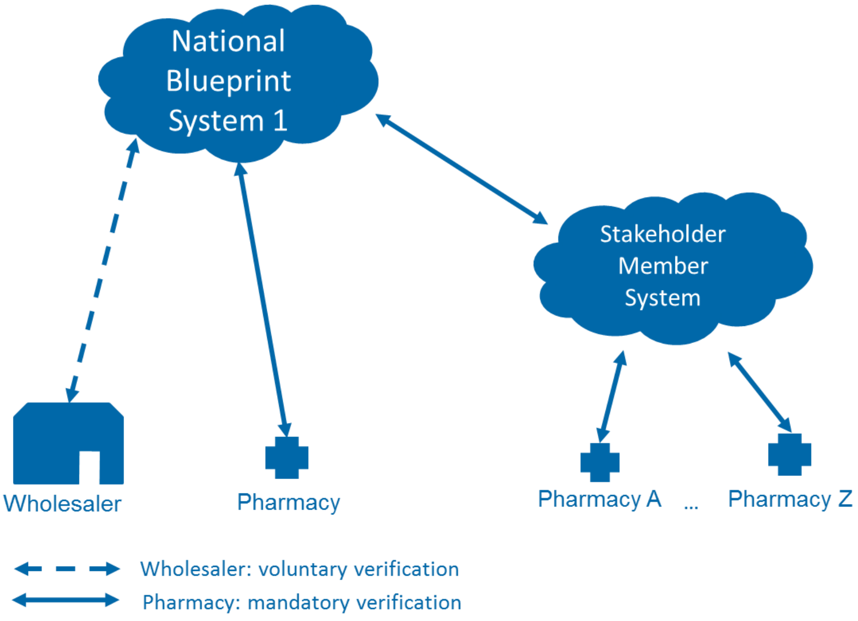

National Blueprint System 1
Stakeholder Member System
Wholesaler
Pharmacy
Pharmacy A
...
Pharmacy Z
Wholesaler: voluntary verification
Pharmacy: mandatory verification

Figure 5: The different ways to connect

Coding on pack

To verify / dispense / decommission a product the operator needs to scan the data matrix code on the pack. This code is mandatory for every RX product within the European market. The actual scanning procedure will depend on the circumstances and the installed software used in the particular pharmacy.

GTIN definition

- *Definition of a GTIN*
  The Global Trade Item Number™ (GTIN™) is used for the unique identification of trade items worldwide.
  GTINs may be 8, 12, 13 or 14-digits in length.

Their data structures require up to 14-digit fields, and **all GTIN processing software should allow for 14 digits**.
GTIN-14 is a 14-digit number used to identify trade items at various packaging levels.

- *GTINs in Healthcare*
  Global Trade Item Numbers (GTINs) uniquely identify items that are traded (pharmaceuticals, medical devices, etc.) in the Supply Chain. A change to one aspect, characteristic, variant or formulation of a trade item may require the allocation of a new GTIN.
  Brand Owners who hold the specifications of a healthcare item must properly allocate and maintain their GTINs to enable trading partners to distinguish products effectively for regulatory, supply chain and patient safety concerns.

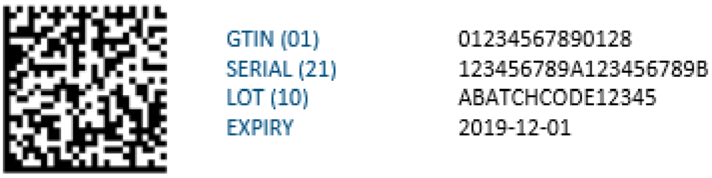


Figure 6: GTIN Coding example

In the figure above, the data matrix code to be scanned is on the left side. An exemplary human readable information is on the right side. The identifier at the beginning of a line can change. In this example, we have the GS1 coding schema. That means

(01) = GTIN (global trade item number)

(21) = serial number

(17) = expire date – same like S.K.T:

(10) = batch number – same like LOT:

PPN definition

The Pharmacy Product Number (PPN) has been developed by a stakeholder consortium consisting of pharmaceutical product manufacturers, distributors and pharmacists in order to integrate existing local systems into global standards of the International Standardization Organization (ISO) and ensuring the PPN is internationally unique. IFA acts as the executive organization maintaining the PPN.

A PPN (Pharmacy Product Number) has the following structure:

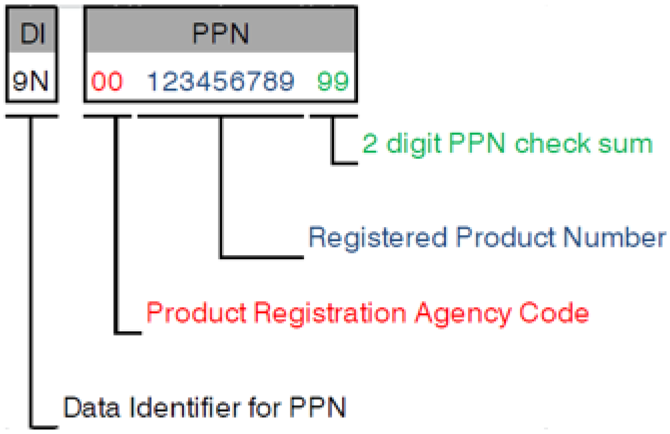
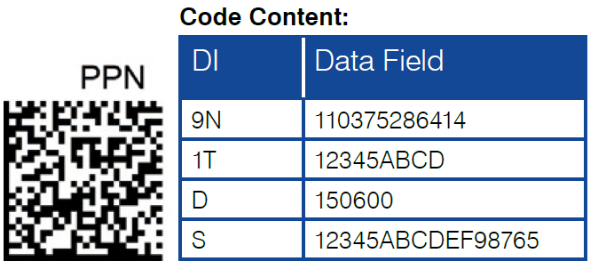


Figure 7: PPN structure

- *DI (Data Identifier)* is a prefix to the actual PPN. It is used to indicate the beginning of a PPN in data structures such as data matrix codes.
- *Product Registration Agency (PRA) Code* indicates the type of Registered Product Number that follows. E.g. 11 signifies a German PZN (Pharmazentralnummer).
  See also following slide.
- *Registered Product Number* is the embedded product number, e.g. a German PZN, an Austrian PZN or a GTIN.
- PPN Check Sum is a 2 digit checksum, ensuring a high level of security against typing or reading errors.



Code Content:

DI	Data Field
9N	110375286414
1T	12345ABCD
D	150600
S	12345ABCDEF98765

Figure 8: PPN coding example

Use Cases

To handle the different use cases in the distributor’s daily work different web services are implemented.

The National Medicines Verification System has different clusters of use cases that are linked to dedicated web services.

For the connection to the distributor the G1xx services are the main group and will be described in this document. For administrative activities some services in the G4xx range will be used.

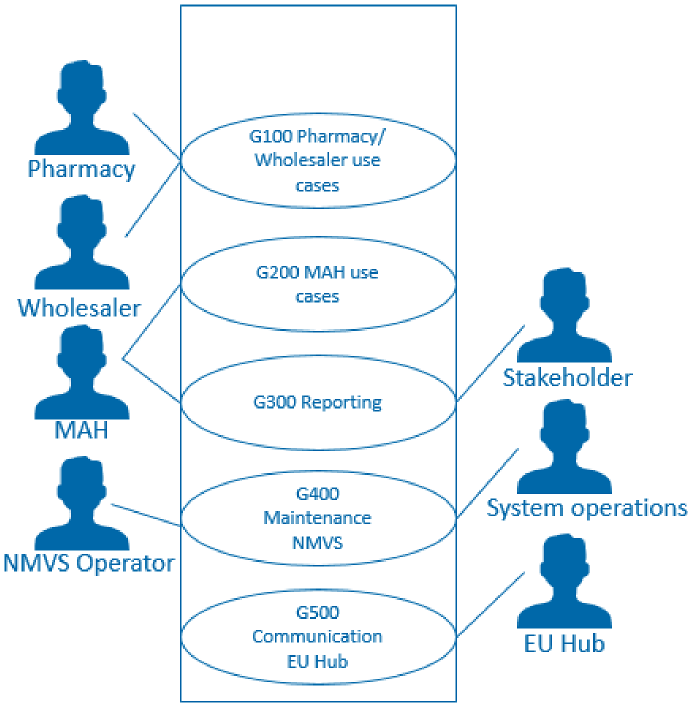


Figure 9: Web Services range

Synchronously vs. Asynchronously

There are two different ways of communication. The first one – and the most suitable one for any activity in the pharmacy - is the synchronous single transaction. The operator at the pharmacy is scanning the data matrix code from a package. The pharmacy IT system is submitting the information with the respective web service to the NMVS. The result of the service request will be submitted directly after processing back to the pharmacy system and will be displayed to the operator. A typical process would be the dispensing of a product at the desk to a patient / customer. The cashier system will scan the code as usually during the sales process. The software of the cashier system that is part or connected to the pharmacy software is processing this information plus additional pieces of information and submitting these to the NMVS. The NMVS will check if the product is valid for dispensing, change the status of the pack to 'dispensed' and inform the pharmacy software about the result. The pharmacist will get the result displayed on the cashier system screen during the sales process.

The second way of interfacing is the asynchronous bulk or the so called offline connection. The offline connection will be the procedure that is to be used if connection issues occur and a single synchronous service is not available. In that case, the operator is able to scan the different packages and the pharmacy software stores the requests in an internal queue. After a successful reconnection to the NMVS, the pharmacy software sends out the stored events to the NMVS. The correct order is crucial for this process. As there might be interdependencies between the different scans the submission of the events must be transmitted in the original order. The results of the different requests will not come in separate statements. There will be only one combined answer with all the results for all called service requests.

Another way to use the asynchronous bulk connection is mainly linked to the wholesaler processes. There is a possibility to link a set of scans to one request. This will only be possible if all scans belong to the same product code. The bulk activity is – other than the offline scans – linked to one activity. This could be a ‘verify’ for more than one package out of the same product code. The wholesaler software will collect all the scans and submit the serial numbers in one web service request to the NMVS. As this is an asynchronous process, too, the wholesaler system should not wait for results. The answer to the request will come as a bulk answer.

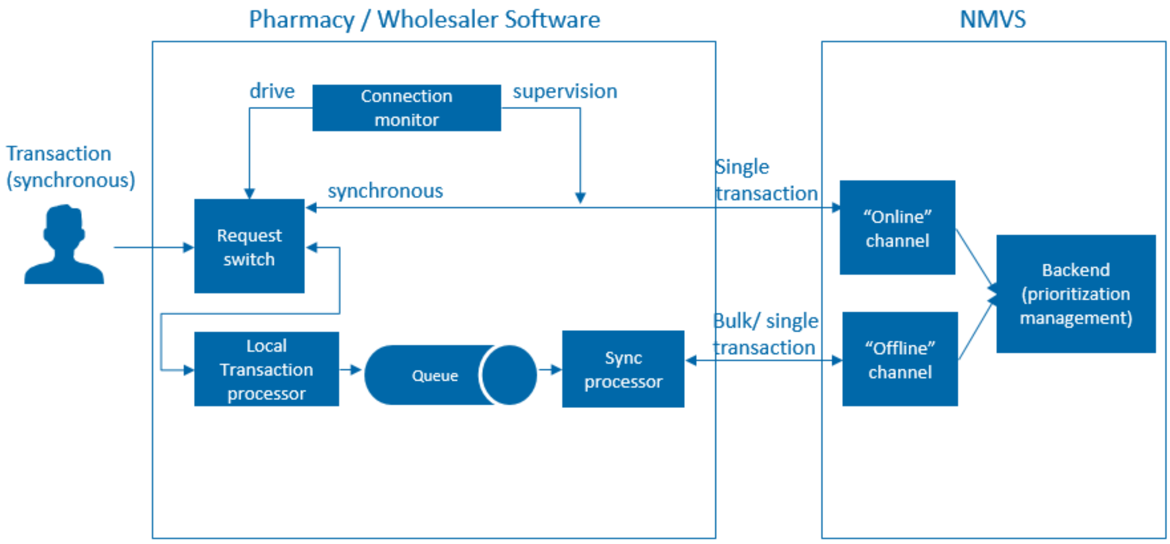


Figure 10: synchronously vs. asynchronously

The following chapter describes in detail the communication interface between a distributor in a country and the National Medicines Verification System. The software supplier for the particular distributor’s IT system needs to implement the interface according to the following description.

The structure of the documentation is based on use cases. There are services that are triggered by the daily work of the stakeholder operator as well as administrative services that have to be triggered by the used distributor’s IT system. The following table should be used as an overview. Not all of them are allowed to be used by each stakeholder group.

Permission Mapping

Process ID	Title	Wholesaler	Pharmacy/ Hospital
G100 - Distributor Transactions			
Single Pack Transactions			
G110	Verify single pack	Yes	Yes
G112	Manual Verification of Single Pack	No	No
G120	Dispense single pack	Yes	Yes
G122	Dispense single pack manual entry	Yes	Yes
G130	Destroy single pack	Yes	Yes
G132	Manual Destruction of Single Pack	No	No

G140	Export single pack	Yes	No
G142	Manual Export of Single Pack	No	No
G150	Sample single pack	Yes	Yes
G152	Manual Set Sample of Single Pack	No	No
G160	Free sample single pack	Yes	No
G170	Lock single pack	Yes	No
G172	Manual Locking of Single Pack	No	No
G121	Undo dispense single pack	Yes	Yes
G141	Undo export single pack	Yes	No
G151	Undo sample single pack	Yes	Yes
G161	Undo free sample single pack	Yes	No
G171	Undo lock single pack	Yes	No
G180	Stolen single pack	Yes	No
G182	Stolen single pack manual entry	Yes	No
Homogenous Bulk Transactions			
G115	Bulk verify packs	Yes	No
G125	Bulk dispense packs	Yes	No
G135	Bulk destroy packs	Yes	No
G145	Bulk export packs to Non-EU	Yes	No
G146	Bulk check out packs for EU distribution	No	No
G155	Bulk sample packs	Yes	No
G165	Bulk free sample packs	Yes	No
G175	Bulk lock packs	Yes	No
G185	Bulk stolen packs	Yes	No
G127	Bulk undo dispense packs	Yes	No
G147	Bulk undo export packs to Non-EU	Yes	No
G148	Bulk undo check out packs for EU distribution	No	No
G157	Bulk undo sample packs	Yes	No
G167	Bulk undo free sample packs	Yes	No
G177	Bulk undo lock packs	Yes	No
G188	Request bulk transaction result	Yes	No
Mixed Bulk Transactions			
G195	Submit mixed bulk transaction	No	Yes
G196	Request mixed bulk transaction result	No	Yes
Others			
G199	Request pending bulk transaction ids	Yes	Yes
G101	Download product master data from NMVS	Yes	Yes

G400 - Administration ***Various functionalities to manage access to/usage of the NMVS plus administration of product master data***			
G445	Change password	Yes	Yes

G482	Load Terms and Conditions	Yes	Yes
G483	Confirm Terms and Conditions	Yes	Yes
G600 - Stakeholder Onboarding Support ***Onboarding support for stakeholders via NMVS portal***			
G615	Download client certificate PKI	Yes	Yes

Table 3: Web Service overview

7.3 NMVS – Blueprint Release Roadmap

The EU legislation EU/2011/62 („Falsified Medicines Directive“) with the local adaption and the specific local requirements is still in adjustments. The NMVS specification is in finalization but the software needs to be adopted. To provide the needed functionality and to provide the software suppliers the needed test environment, the NMVS will appear in four major releases.

The availability dates for the productive systems are:

- Core 1.2 – general availability 11/2018 - Blueprint, except for certain reports.

For test purposes a country specific integration system will be available approx. two weeks earlier. To provide an additional service to all software suppliers a beta version of the latest core version will be available approximately eight weeks earlier. Note: The beta version is tested by our development engineers. In parallel, the Q&A department will run extensive test procedures on the beta release. Thus, it could be possible that you will detect some bugs or features within that period.

The covered functionalities of the four releases are described in the next chapters.

7.3.1 Blueprint Functionality – Release - Core 1.1

In the second release additional use cases will be implemented

Supported use cases:

- Handling of multi market packs
- Web service – download master data for pharmacy/wholesaler – G101
- Use cases
    - DESTROY
    - STOLEN
    - SAMPLE
    - FREE SAMPLE
    - EXPORT
    - LOCKED

7.3.2 Blueprint Functionality – Release - Core 1.2

With regards to the stakeholder connectivity, the main focus lays on the implementation of alert reporting.

Administrative functions:

- Handling of intermarket transactions
- Product Withdrawal
- Reporting of alerts (exceptions) to EU-hub

7.4 Structure of the use case description

The structure of each service description is similar. Following sub chapters are part of the description:

- Abstract of the service
- Process description
- Use Case Diagram
- Detailed process description
- Web service description
- Web service input parameter
- Web service output parameter
- Web service detailed description of the parameter

Abstract of the service:

Short description to introduce the service.

Process description:

Explanation of the different steps of the service in the 'good' case from the first action up to the out coming result.

Use Case Diagram:

Diagram of the service and the involved layer in swim-lane systematic. The drawing includes the good cases as well as all error paths. The diagram demonstrates the interaction between the different layers. It will not demonstrate the functional step inside the distributor's IT systems.

Detailed process description:

Detailed explanation of the service step-by-step including all error cases and results.

Web service description:

Explanation of the web service that needs to be used for that specific process.

Web service input parameter:

Description of the input parameters in detail for that specific service.

Web service output parameter:

Description of the output parameters in detail for that specific service.

Web service detailed description of the parameter:

Description of various parameter formats with examples.

7.5 Process Extension for Multi-Market Packs

Some medicine products are manufactured as multi-market packs. These are packs which can be legitimately sold within multiple markets by the use of the same product code (in general not the same reimbursement code). The national system where the pack is scanned is responsible to initiate the synchronization process with the European Hub.

Multi-market packs will have no influence on the business processes within the distributors' IT systems. Therefore, this documentation will not go into detail of the description of the multi-market procedures in the NMVS.

8 Technical Overview

8.1 Architecture

The NMVS API distributors interface is based on standards generally known as web services which include the Simple Object Access Protocol (SOAP), Web Services Definition Language (WSDL), and the XML Schema Definition language (XSD). A wide range of development tools on a variety of platforms support web services. It is possible to use any programming language supporting web services to integrate with the NMVS API.

8.2 Connection

Connection initiation starts from the client side to the NMVS. The NMVS is only accessible through HTTPS. Clients must comply with TLS version 1.2 in order to connect with the API.

The NMVS API has been implemented to comply with the following specifications:

Standard	Website
Simple Object Access Protocol (SOAP) 1.2	https://www.w3.org/TR/soap12/
Web Service Description Language (WSDL) 1.1	https://www.w3.org/TR/2001/NOTE-wsdl-20010315
WS-I Basic Profile 1.1.1	http://www.ws-i.org/Profiles/BasicProfile-1.1-2006-04-10.html
Extensible Markup Language (XML)	http://www.w3.org/XML/

Table 4: Standards compliance

These standards are mandatory for the communication with the NMVS API.

8.2.1 HTTPS

The NMVS API is only accessible through HTTPS. Clients must comply with TLS version 1.2 in order to communicate with the NMVS.

For every transaction communication clients should implement HTTP keep-alive to obtain persistent connections. For that purpose, NMVS provides a special SOAP Request. Please refer to section 8.2.5.5

Cipher Suites supported by NMVS
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

Table 5: Supported cipher suites

Algorithms explicitly excluded from Cipher Suites
!EDH-aRSA
!aNULL
!eNULL
!LOW
!3DES
!MD5
!EXP
!SRP
!DSS
!RC4
!DES
!RC2
!EXPORT
!PSK
!SEED
!ECDSA
!AHD
!IDDEA

Table 6: Excluded algorithms

8.2.2 Client Certificates

The NMVS provides X.509 certificates. Any client needs to authenticate through a valid certificate installed in the local trust store of the connecting client.

Please refer to chapter 15 “Security” for further details.

8.2.3 SOAP

As mentioned above, SOAP V.1.2 is mandatory for any communication with the API. The NMVS API uses document-literal SOAP messaging, not rpc-encoding, rpc-literal nor document-encoding. For document-literal, a single service interface call passes an XML document with the request to the NMVS API server which responds with a XML document instance.

As the SOAP requests and the responses of the NMVS API follow current standards, any programming language with the appropriate library support can be used. Languages known to have this support include C# and Java (among others).

8.2.4 Internationalization and Character Sets

The NMVS API assumes that all request data is in Unicode Transformation Format, 8-bit encoding form (UTF-8).

8.2.5 Connection Handling

This chapter defines connection handling requirements in more detail.

8.2.5.1 DNS

All technical endpoints have registered DNS names.

8.2.5.2 Client Certificate Validation

Before any connection attempt, the client software must ensure that the client certificate used for authentication is valid. This means the end-to-end certificate chain is valid and its validity period is not expired.

8.2.5.3 Rejected Connection Attempts

In case that a connection attempt is rejected from the NMVS which will lead to a returning http status (code 403), it is not allowed to continue with further connection attempts. The status code may indicate an invalid client certificate that needs to be manually checked on client side.

8.2.5.4 Latency / Keep alive

In order to reduce latency (important for single pack transactions at the point of dispense) it is advised to reuse an already established (http keep alive header) connection to the NMVS as the initial connection setup is time-consuming compared to the pure processing time of the transactions. For this purpose all services support the HTTP 1.1 keep alive mechanism. The timeout is set to 75 seconds. So for subsequent transactions in combination with low bandwidth (and high latencies) this can speed up transaction times.

8.2.5.5 Ping Service

All NMVS web services provide a simple ping service to test the general connectivity to the NMVS. The service can also be used for holding connections to prevent connection timeouts.

8.2.5.6 Connection Termination

The client should drop connections if no activity is expected in a reasonable amount of time. Possibly immediately after a transaction has been executed.

8.2.5.7 Reconnection Policy

If connection attempts to the NMVS fail, wait times between connection attempts must be added as the number of failed attempts increases. After each failed attempt, the waiting time should be increased by e.g. 30 seconds.

8.2.5.8 Parallelism

In general, a client should maintain not more than one connection to each service endpoint of the NMVS. In case of higher parallelism on synchronous services it is possible to open additional connections. For services with asynchronous processing it is not allowed to submit requests in parallel as processing order cannot be guaranteed in that case.

8.3 List of Web Services

The WSDL files for web services of the distributors interface and its underlying base and catalogue types – XSD files – are required for accessing and developing with the NMVS API.

With Core 1.2 there is a version 3 of the interface. It can be accessed via new endpoints that are updated in the SWS portal. The previous interface (version 2) remains unchanged. Version 1 of the interface is removed since Core release 1.2 and can no longer be called.

File	Description
WS_SINGLE_PACK.wsdl	Web services to submit single pack transactions.
WS_BULK.wsdl	Web services to submit a bulk of packages in one transaction and to get the result. All packages in one request belong to one batch. Different batches in one request are not allowed.
WS_MIXED_BULK.wsdl	Web services to transmit transactions that are built-up while the NMVS is not reachable in order to get the result of processing. One request may contain different single transaction requests.
WS_MASTER_DATA.wsdl	Web services to get all products from NMVS.
WS_SUPPORT.wsdl	Web services for support functions e.g. changing passwords.
WS_PKI.wsdl	Web services for PKI operations e.g. to retrieve the client certificate. This service does not require a client certificate itself to be invoked.

Table 7: List of web services

8.3.1 List of Web Services – Version 1

The following are the locations and descriptions of the WSDL and XSD files for Version 1 (V1).

NOTE: With core release 1.2 the version 1 of the web services will be obsolete and removed. Please Upgrade to a newer version

8.3.2 List of Web Services – Version 2

The following are the locations and descriptions of the WSDL and XSD files for Version 2 (V2).

NOTE: For version 2 please refer to NMVS - WSDL - XSD_V2.zip

8.3.3 List of Web Services – Version 3

The following are the locations and descriptions of the WSDL and XSD files for Version 3 (V3).

NOTE: For version 3 please refer to NMVS - WSDL - XSD_V3.zip

8.3.4 Web Service Life Cycle

With NMVS Release 1.02 arvato will introduce the web service life cycle for the first time. The NMVS will, from then on, only support the latest two versions of the web service. With the release 1.03 this will be version 2 & 3.

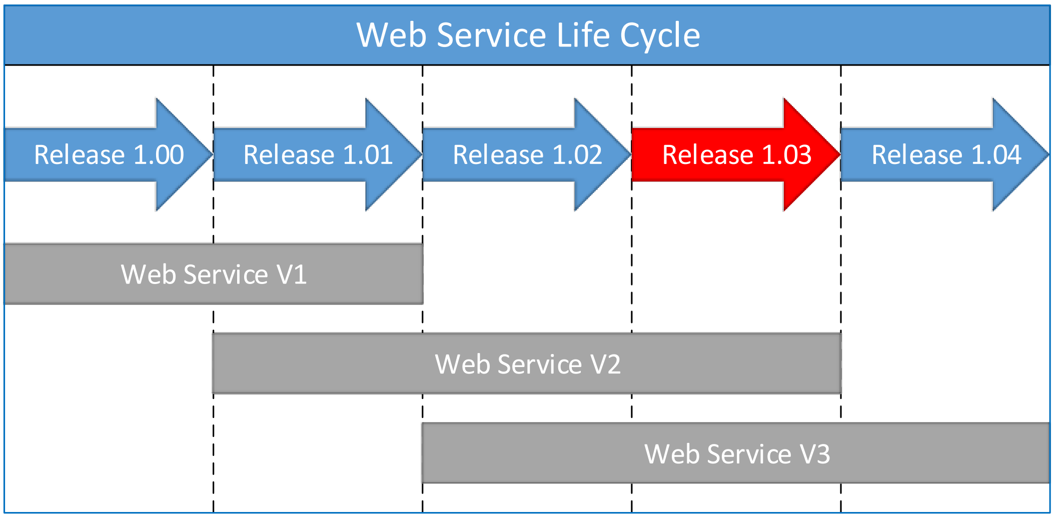


Figure 11: Web service life cycle

Not every NMVS release will come with a new version of the web service. But if it´s the case, it will be clearly communicated in advance.

NOTE: With NMVS release 1.03 there are no changes regarding the interface versions. This means, V2 and V3 will remain active and V3 is still the newest interface version.

8.3.5 Structure of the WSDL / Scheme

The WSDLs contain all definitions regarding messages, port types, bindings, ports and services. Only the types are split into different files. The file “NMVS_Base_Types.xsd” contains the definitions of the simple types with length, restrictions and description. The file “NMVS_Catalog_Types.xsd” contains the enumerations, for example the error codes of NMVS. The file “NMVS_Composite_Types.xsd” is based on the types of these former files and contains all complex types that are needed for more than one input or output structure.

The other files contain one input or one output structure and those types that are used by them exclusively. All input structure file names start with “I” and all output structures start with “O”.

8.3.6 List of Endpoints

To perform transactions with the NMVS in the Sandbox environment, the below listed endpoint URLs have to be used. These are currently available in interface version 2 and 3.

Endpoint Description	Endpoint URL	IP	TTL
Single transactions: e.g. verification, dispense, undo, etc.	https://ws-single-transactions-int-bp.nmvs.eu:8443/WS_SINGLE_TRANSACTIONS_V1/SinglePackServiceV20	146.185.100.16	86400

Endpoint Description	Endpoint URL	IP	TTL
Bulk transactions: e.g. bulk verification, bulk dispense	https://ws-bulk-transactions-int-bp.nmvs.eu:8445/WS_BULK_TRANSACTIONS_V1/BulkServiceV20	146.185.100.16	86185
Mixed bulk transactions	https://ws-mixed-bulk-transactions-int-bp.nmvs.eu:8446/WS_MIXED_BULK_TRANSACTIONS_V1/MixedBulkServiceV20	146.185.100.16	86266
Master data transactions	https://ws-master-data-int-bp.nmvs.eu:8447/WS_MASTER_DATA_V1/MasterDataServiceV20	146.185.100.16	86331
Support transactions: Terms & Conditions, password change,	https://ws-support-int-bp.nmvs.eu:8448/WS_SUPPORT_V1/SupportServiceV20	146.185.100.16	86400
Client certificate	https://ws-pki-int-bp.nmvs.eu:8444/WS_PKI_V1/PkiServiceV20	146.185.100.16	86090

Table 8: List of endpoints for NMVS Sandbox environment version 2

The difference between interface version 2 and 3 are the version numbers at the end of the endpoint URL. These are https://.../WS_SINGLE_TRANSACTIONS_V1/SinglePackService**V20** and https://.../WS_SINGLE_TRANSACTIONS_V1/SinglePackService**V30**.

Endpoint Description	Endpoint URL	IP	TTL
Single transactions: e.g. Verification, Dispense, Undo, etc.	https://ws-single-transactions-int-bp.nmvs.eu:8443/WS_SINGLE_TRANSACTIONS_V1/SinglePackServiceV30	146.185.100.16	86400
Bulk transactions: e.g. bulk verification, bulk dispense	https://ws-bulk-transactions-int-bp.nmvs.eu:8445/WS_BULK_TRANSACTIONS_V1/BulkServiceV30	146.185.100.16	86185
Mixed bulk transactions	https://ws-mixed-bulk-transactions-int-bp.nmvs.eu:8446/WS_MIXED_BULK_TRANSACTIONS_V1/MixedBulkServiceV30	146.185.100.16	86266
Master data transactions	https://ws-master-data-int-bp.nmvs.eu:8447/WS_MASTER_DATA_V1/MasterDataServiceV30	146.185.100.16	86331

Endpoint Description	Endpoint URL	IP	TTL
Support transactions: Terms & Conditions, password change,	https://ws-support-int-bp.nmvs.eu:8448/WS_SUPPORT_V1/SupportServiceV30	146.185.100.16	86400
Client certificate	https://ws-pki-int-bp.nmvs.eu:8444/WS_PKI_V1/PkiServiceV30	146.185.100.16	86090

Table 9: List of endpoints for NMVS Sandbox environment version 3

8.3.7 Data Types

An entire list of all data types can be found in “**Interface Description for Distributors.xls” in the related interface version** at the sheet **“Data Type”**

The table is structured next to the general setup of an XML. Thus, it starts with all data types for the Header, followed by all types for the User Software, the Transaction information and so on.

To avoid deviant information between both documents we only show an example of listed data types from the interface Description. Please use only the Sheet Data Type in the Interface Description for local developments.

Attributes	Description
A – Header data	
A1 - Authentication	
Client ID	Identification of a client. A client unites a group of users (e.g. pharmacies, wholesalers, hospitals etc.). In case of a centralized access scenario (e.g. all hospitals are connected via an concentrator - i.e. a central server), a client may only have one user, i.e. the central server itself, which requires that the local NMVO tolerates the use of concentrators in a particular country.
User ID	Identification of an individual user (i.e. a member of the client).
Password	Password to authenticate a user
Sub-User ID	The Sub User ID is mandatory in case an concentrator system is in place, which requires that the local NMVO tolerates the use of concentrators in a particular country. In this case, the Sub User ID provides a pseudonymous, unique ID of the (otherwise invisible) users behind the concentrator system (e.g. the hospitals).

Attributes	Description
	NOTE: The Sub User ID is an optional field. If this information is not necessary for your implementation, it can be commented out. The length of the Sub User ID was reduced to 35 characters
A2 – User software	
Supplier	Vendor of the client/stakeholder software ID that was validated during the onboarding process (e.g. “PharmaSoftProvider”)
Product	Client/stakeholder software ID that was validated during the onboarding process (e.g. “PharmaSoft”)
Software version	Attribute to identify the version of the client/stakeholder system. The version ID of the stakeholder software ID that was validated during the onboarding process (e.g. “PharmaSoft 2.4”)
A3 – Transaction data	
Language code	Language code, ISO-639-3 compliant. Standard fallback language is English (eng). Supported languages include eng, deu. Further languages may be supported after consultation with the respective NMVO.
Client transaction ID	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.
NMVS transaction ID	Unique identifier of the transaction from the NMVS.
NMVS timestamp	Processing timestamp of the NMVSin UTC, marks the end of processing within the NMVS.
Request type	Type of request.
A4 – Notifications (only in response)	
Notification code	Alphanumeric notification code, e.g. NMVS_NOTIFY_AU_01.
Notification description	Textual description of the notification code, e.g. reminder of password expiration.

Table 10: Data Types in the Interface Description

8.3.8 Catalogue / Enumerations

The response of the National Verification System contains in some attributes standard or catalog values. Detailed pieces of information are given for the attributes PACK STATE, PACK-RELATED-REASON, RETURN CODE, PROCESSES and PRODUCT SCHEME.

An entire list of the catalogue can be found in “**Interface Description for Distributors.xls”** in the related interface version at the sheet **“Catalogue”**

To avoid deviant information between both documents we only show an example of listed catalog values from the interface description. Please only use the Sheet Catalog in the interface description for local developments.

Attributes	Detailed Information	Description
B – pack data		
Pack state	ACTIVE	Pack can be decommissioned
	INACTIVE	Pack cannot be decommissioned
Pack-related reasons	SUPPLIED	Pack is supplied to the public
	DESTROYED	Pack is destroyed. This pack state reason is a final state and cannot be undone
	LOCKED	Pack is locked. This pack state reason can be overwritten by “destroyed” “stolen”
	EXPORTED	Pack is exported outside the EU
	SAMPLE	Pack was given away as a sample for national authorities
	STOLEN	Pack is marked as stolen. This pack state reason is a final state and cannot be undone
	CHECKED_OUT	Pack is repacked into a new pack
	FREESAMPLE	Pack is given away as a free sample
	RECALLED	Batch is recalled and is not allowed to be dispensed
	EXPIRED	Batch is expired and is not allowed to be dispensed
	WITHDRAWN	Product is inactive with the property withdrawn and not allowed to be dispensed
	UNKNOWN	Product, batch or pack is unknown and not allowed to be dispensed

Table 11: Extract from Catalogue

8.3.9 Mapping Services to Use Cases

Service name	Use case code	Business-Process
G100 – Distributor Transactions		
Single Pack Transactions – Do-Transactions		
WS_SINGLE_PACK	G110	Verify single pack
WS_SINGLE_PACK	G120	Dispense single pack
WS_SINGLE_PACK	G122	Dispense single pack manual entry
WS_SINGLE_PACK	G130	Destroy single pack
WS_SINGLE_PACK	G140	Export single pack
WS_SINGLE_PACK	G150	Sample single pack
WS_SINGLE_PACK	G160	Free sample single pack
WS_SINGLE_PACK	G170	Lock single pack
WS_SINGLE_PACK	G180	Stolen single pack
WS_SINGLE_PACK	G182	Stolen single pack manual entry
Single Pack Transactions – Undo-Transactions		
WS_SINGLE_PACK	G121	Undo dispense single pack
WS_SINGLE_PACK	G131	Undo destroy single pack (Not usable as not permitted by the URS)
WS_SINGLE_PACK	G141	Undo export single pack
WS_SINGLE_PACK	G151	Undo sample single pack
WS_SINGLE_PACK	G161	Undo free sample single pack
WS_SINGLE_PACK	G171	Undo lock single pack
WS_SINGLE_PACK	G181	Undo stolen single pack (Not usable as not permitted by the URS)
Homogenous Bulk Transactions – Do-Transactions		
WS_BULK_TRANSACTIONS	G115	Bulk verify packs

Service name	Use case code	Business-Process
WS_BULK_TRANSACTIONS	G125	Bulk dispense packs
WS_BULK_TRANSACTIONS	G135	Bulk destroy packs
WS_BULK_TRANSACTIONS	G145	Bulk export packs to Non-EU
WS_BULK_TRANSACTIONS	G155	Bulk sample packs
WS_BULK_TRANSACTIONS	G165	Bulk free sample packs
WS_BULK_TRANSACTIONS	G175	Bulk lock packs
WS_BULK_TRANSACTIONS	G185	Bulk stolen packs
Homogenous Bulk Transactions – Undo-Transactions		
WS_BULK_TRANSACTIONS	G127	Bulk undo dispense packs
WS_BULK_TRANSACTIONS	G147	Bulk undo export packs
WS_BULK_TRANSACTIONS	G157	Bulk undo sample packs
WS_BULK_TRANSACTIONS	G167	Bulk undo free sample packs
WS_BULK_TRANSACTIONS	G177	Bulk undo lock packs
Homogenous Bulk Transactions – Other-Transactions		
WS_BULK_TRANSACTIONS	G188	Request bulk transaction result
Mixed Bulk Transactions		
WS_MIXED_BULK	G195	Submit mixed bulk transaction
WS_MIXED_BULK	G196	Request mixed bulk transaction result
Other		
WS_BULK_TRANSACTIONS	G199	Request pickup IDs for bulk transaction result
WS_MASTER_DATA	G101	Download product master data from NMVS
G400 – Administration		
WS-SUPPORT	G445	Change Password
G600 – Stakeholder Onboarding Support		

Service name	Use case code	Business-Process
WS_PKI	G615	Download client certificate PKI

Table 12: Use cases of services

8.4 General Request / Response Structure

All requests and responses are embedded in the SOAP body. The first level contains the process number and the kind of element (request or response). The child elements are the header and the body.

With Core 1.2 there is a new version for web services V3 in addition to the existing V2, which specifies which of the two versions is to be used, while the response has been extended by the characteristic Notification. The V2 and V3 versions are also used by Core 1.3. Example below: Response V2.

Within the interface description an example for each web service is available.

```
<G110Request>
   <Header>
      <Auth>
         <ClientLoginId>G4321-14-01RT</ClientLoginId>
         <UserId>9821</UserId>
         <Password>838hdjLk#</Password>
         <!--Optional:-->
         <SubUserId>9824</SubUserId>
      </Auth>
      <UserSoftware name="EasyPharm" supplier="PharmSoft Ltd." version="3.2.0"/>
      <Transaction>
         <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
         <Language>deu</Language>
      </Transaction>
   </Header>
   <Body>
      <Product>
         <ProductCode scheme="GTIN">05060141900015</ProductCode>
         <Batch>
            <Id>1264</Id>
            <ExpDate>171200</ExpDate>
         </Batch>
      </Product>
      <Pack sn="XVTR75973491006155"/>
   </Body>
</G110Request>
```

Figure 12: Example Request

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
   <soap:Body>
      <ns2:G110Response xmlns:ns1="urn:types.nmvs.eu:v2.0"
xmlns:ns2="urn:wsdltypes.nmvs.eu:v2.0">
         <ns1:Header>
            <ns1:Auth>
               <ns1:ClientLoginId>TESTNOTI</ns1:ClientLoginId>
               <ns1:UserId>USERG110</ns1:UserId>
            </ns1:Auth>
            <ns1:Transaction>
               <ns1:Language>deu</ns1:Language>
               <ns1:ClientTrxId>trx-1-2-3-110</ns1:ClientTrxId>
               <ns1:NMVSTrxId>3c77dcb7c7ad481c9c6cddaad1a79a1b</ns1:NMVSTrxId>
               <ns1:Timestamp>2018-03-09T15:03:10.444+01:00</ns1:Timestamp>
            </ns1:Transaction>
            <ns1:Notification ns1:notificationCode="NMVS_NOTIFY_AU_01" ns1:notificationDesc="You are using an old password.
The password of your user account has changed, so please adjust your password as soon as possible.
The password you are using will be deactivated in $1 day(s).
"/>
         </ns1:Header>
         <ns1:Body>
            <ns1:Product>
               <ns1:ProductCode ns1:scheme="PPN">110258282253</ns1:ProductCode>
               <ns1:Batch>
                  <ns1:Id>U0226</ns1:Id>
                  <ns1:ExpDate>111111</ns1:ExpDate>
               </ns1:Batch>
            </ns1:Product>
            <ns1:Pack ns1:sn="E0XG32FRM57GGSJZ4AMG" ns1:state="UNKNOWN"/>
            <ns1:ReturnCode ns1:code="NMVS_NC_PC_01" ns1:desc="Unknown product code."/>
         </ns1:Body>
      </ns2:G110Response>
   </soap:Body>
</soap:Envelope>
```

Figure 13: Example Response V2

8.4.1 Header

The header contains the authentication information, information related to the software that requests the NMVS service and transaction IDs for identifying the transaction.

8.4.1.1 Request Header

8.4.1.1.1 Authentication Data

The authentication element under the header node contains all necessary information to log in to the NMVS. It is important to know how the requesting system connects to NMVS. In case of a concentrator system that provides pseudonymous connection, the sub user ID is mandatory, otherwise it is optional and will be ignored. In case of undo operations in a direct scenario, NMVS will check if the user ID is the same as the one who has performed the preceding DO-operation (e.g. G120 – Dispense Single Pack and G121 Undo Dispense Single Pack). In a pseudonymous integration scenario, the sub user ID is checked, if it is the same as the preceding do-operation. For more information about integration scenarios, please refer to the corresponding chapter in this document.

```
<Auth>
   <ClientLoginId>G4321-14-01RT</ClientLoginId>
   <UserId>9821</UserId>
   <Password>838hdjLk#</</Password>
   <!--Optional:-->
   <SubUserId>9824</SubUserId>
</Auth>
```

Figure 14: Example Authentication Data

NOTE: The length of the Sub User ID was reduced to 35 characters. There is an optional attribute within the <auth> tag. The "authMethod" attribute is currently not used and can be ignored. It might be used in future versions of the interface.

8.4.1.1.2 User Software Data

The user software data describes the software that is sending the requests to NMVS. The information may help the NMVS-Provider to identify insufficient implemented software which causes some errors.

```
<UserSoftware name="EasyPharm" supplier="PharmaSoft Ltd." version="3.2.0"/>
```

Figure 15: Example User Software Data

8.4.1.1.3 Transaction Data

The transaction data includes a unique transaction ID for the request which is sent to the NMVS.

```
<Transaction>
   <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
   <Language>deu</Language>
</Transaction>
```

Figure 16: Example Transaction Data

8.4.1.2 Response Header

8.4.1.2.1 Response

The response specifies the interface version to be used.

```
<ns2:G110Response xmlns:ns1="urn:types.nmvs.eu:v2.0" xmlns:ns2="urn:wsdltypes.nmvs.eu:v2.0">
```

Figure 17: Example Response V2

8.4.1.2.2 Authentication Data

The authentication element contains the user login data.

Example:

```
<Auth>
```

```
    <ClientLoginId>G4321-14-01RT</ClientLoginId>
    <UserId>9821</UserId>
    <!--Optional:-->
    <SubUserId>9824</SubUserId>
</Auth>
```

Figure 18: Response Authentication Data

8.4.1.2.3 Transaction Data

The transaction data contains the (external) client transaction ID that NMVS has received with the request and the NMVS adds its own (internal) transaction ID and the timestamp when the request has been processed.

Example:

```
<Transaction>
    <Language>deu</Language>
    <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
    <NMVSTrxId>c295a450-0caa-4cf1-7e62-c6865d41</NMVSTrxId>
    <Timestamp>2017-12-31T12:00:00</Timestamp>
</Transaction>
```

8.4.1.2.4 Notification Data

The notification element contains the notification code and a corresponding description. As shown in the example below, this data is used to give the user additional information about his account. The NMVS uses this field, for example, to forward the information about an upcoming expiration of the password. The notifications are supposed to be presented to the user so he can act properly.

Example:

```
<ns1:Notification ns1:notificationCode="NMVS_NOTIFY_AU_01" ns1:notificationDesc="You are us-
ing an old password.
The password of your user account has changed, so please adjust your password as soon as
possible.
The password you are using will be deactivated in $1 day(s).
"/>
```

8.4.2 Body

The body of requests contains all input data for processing such as product, batch and pack data. The body of the response contains the processing result. In some cases like errors, it may only contain a return code, in other cases it contains some of the input data, additional data from NMVS, the result of processing and a return code.

More information concerning the body element is provided in “**Interface Description for Distributors.xls”** in the related interface version at the sheet **“Catalogue”**

8.4.3 Error Structure

The general error output are divided in two parts, on the one hand parser errors and on the other hand authentication errors or errors that prevent processing. If the NMVS is not able to read the request because of XML errors, the NMVS will send a SOAP fault. In this case the fault string contains the error code and the error message.

Example:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Body>
      <soap:Fault>
         <faultcode>soap:Client</faultcode>
         <faultstring>NMVS_TE_XM_02: Input data do not match the XML scheme defini-
tion.</faultstring>
      </soap:Fault>
   </soap:Body>
</soap:Envelope>
```

If the authentication fails or the NMVS gets an error during processing the data, an error with the normal response header and return code will be passed back.

Example:

```
<G110Response>
  <Header>
    <Auth>
      <ClientLoginId>G4321-14-01RT</ClientLoginId>
      <UserId>9821</UserId>
      <!--Optional:-->
      <SubUserId>9824</SubUserId>
    </Auth>
    <Transaction>
      <Language>deu</Language>
      <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
      <NMVSTrxId>c234a450-0caa-4cf1-7e62-c6865d41</NMVSTrxId>
      <Timestamp>2017-12-31T12:00:00</Timestamp>
    </Transaction>
  </Header>
  <Body>
    <ReturnCode code="NMVS_FE_AU_01" desc="Unknown combination of user and password" />
  </Body>
</G110Response>
```

In case of an alert, an Alert ID is generated by the NMVS. To transfer the Alert ID to a requesting end user software, the field *Return Code Description* (max. length 250 characters) will be used, which is available in all pack related End User Web Services. The concatenation of the Alert ID with the return code description works as follows:

If the text of the “normal” return code description + Alert ID is > 250 characters, the text of the return code description will be truncated accordingly to the required length, so the complete Alert ID can be mapped into the description field.

- Field content *Return Code Description* with short message text and Arvato style Alert ID:

  Unknown product number AlertID:GB-492d8228-5217-4db0-b912-a16f8a886df0
- Field content *Return Code Description* with short message text and SolidSoft style Alert ID:

  Unknown product number AlertID:IE-2MH-QZ0-A8G-P0A
- Field content *Return Code Description* with long message text and Arvato style Alert ID:

  This is an unnecessarily long and verbose error message describing the fact that the National Medicine Verification System was not able to find in its database the product number that was submitted by the... AlertID:GB-492d8228-5217-4db0-b912-a16f8a886df0
- Field content *Return Code Description* with long message text and SolidSoft style Alert ID:

  This is an unnecessarily long and verbose error message describing the fact that the National Medicine Verification System was not able to find in its database the product number that was submitted by the end user software in... AlertID:IE-2MH-QZ0-A8G-P0A

The length of the Alert ID depends on which type of NMVS generates the Alert ID.

- *Arvato Systems NMVS*
  Alert IDs generated by Arvato Systems contain 39 characters. Their structure is as follows:
  - 2 character Country Code (ISO 3166-2), e.g. **GB**
  - Hyphen „-“
  - 36 character UUID (Universally Unique Identifier, RFC 4122 V4)

  Example: **GB-492d8228-5217-4db0-b912-a16f8a886df0**
- *SolidSoft NMVS*
  Alert IDs generated by SolidSoft contain 18 characters. Their structure is as follows:
  - 2 character Country Code (ISO 3166-2), e.g. **IE**
  - Hyphen „-“
  - Timestamp encoded as 9-character string with two hyphens inserted
  - Hyphen „-“
  - Random number encoded as 3-character string

  Example: **IE-2MH-QZ0-A8G-P0A**

8.4.4 Return and Notification Codes

A complete list of all return and notification codes is listed in the document “TD-001 BP 1.3 ID - G100 Distributor Transactions.xlsx” which can be downloaded from the software supplier portal.

9 Integration Scenarios

Article 13 of the Commission Delegated Regulation (EU) 2016/161 allows reversing the status of a decommissioned unique identifier but expects, among other requirements, the exact identification of the executing instance. Resulting from this requirement, the NMVS API differentiates between three integration scenarios which demands different handling of 'undo' operations. These integration scenarios are:

- direct connection
- pseudonymous connection via concentrator connection
- anonymous connection via concentrator connection

9.1 Direct Connection

The pharmacy directly connects to the NMVS API using a software on its standalone computer.

In this case, all 'undo' services require a check of the ClientLoginID and UserID. Only if both parameters match it will be ensured that the user performing the reverting operation is the same as the user that decommissioned the pack. The undo service will then be executed.

9.2 Concentrator Connection

A concentrator connection is in place when e.g. a pharmacy is connected to the NMVS API via the central system. A central system could be a central pharmacy system managed by a pharmacy association.

Important: Depending on the country you develop client software for, the NMVO of your country might decide not to use or allow concentrators to be used. The concept of a concentrator is described here for the sake of completeness, but it doesn't imply that a concentrator is to be used by all means.

Please contact your local NMVO to clarify how to connect to the NMVS and if you are allowed to set up and use a concentrator.

9.2.1 Pseudonymous Connection

Additionally to the check of the ClientLoginID and UserID, the SubUserID is mandatory in order to perform any 'undo' service.

9.2.2 Anonymous Connection

Additionally to the check of the ClientID and UserID, the RefClientTrxID of the original transaction – the one to be reversed – is needed to perform the 'undo' transaction.

9.3 Bundling of connections

Various distributors will have the need to simultaneously connect multiple computers with the NMVS, for example multiple checkout counters within one pharmacy.

It is highly recommended to bundle all these connections via a central server/proxy and not to connect every single client system separately with the NMVS. That proxy is called cumulator and is nearly the same as a small local concentrator. The only difference is that behind a cumulator all clients share the same user credentials whilst behind a concentrator various clients may reside.

The recommendation is based on multiple problems that might occur with a decentral connection, for example the danger of capacity issues and higher loads on the national system. But the most evident reason is the reproduction of the correct order of events that occur during offline phases.

Once an offline phase occurs clients are to store all service requests locally in order to make up the lack of communication afterwards by sending out all request that could not be sent out before packed in a few mixed bulk requests. For example, if during an offline phase a pack is dispensed (G120) at one checkout counter and returned (G121) at another checkout counter, the pharmacy will have to ensure that – once the internet connection is re-established - those two requests (G120 and G121) will be send out in the original order. That will probably not happen with separate and independent connections from the checkout counters to the NMVS.

The problem can be solved by bundling and passing all requests through a single point of contact, a central server within the pharmacy. The server is then to store and send out the events in the original order.

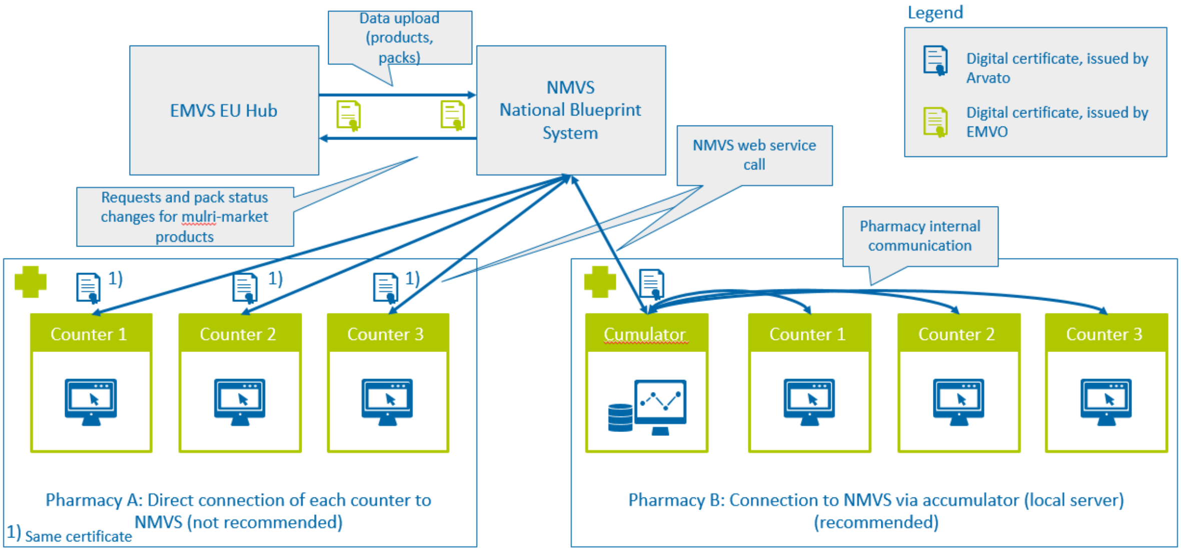

Data upload (products, packs)
EMVS EU Hub
NMVS National Blueprint System
Legend
Digital certificate, issued by Arvato
Digital certificate, issued by EMVO
NMVS web service call
Requests and pack status changes for mulri-market products
Pharmacy internal communication
1)
Counter 1
Counter 2
Counter 3
Cumulator
Counter 1
Counter 2
Counter 3
Pharmacy A: Direct connection of each counter to NMVS (not recommended)
1) Same certificate
Pharmacy B: Connection to NMVS via accumulator (local server) (recommended)

Figure 19: Connection Scenarios for Stakeholders

10 Synchronous Web Service Operations

10.1 Single Pack Transactions

Single pack use cases only operate with one pack. The single pack use cases either return the current pack state or change the state of one pack. The response of a single pack transaction is synchronously returned to the calling client and will inform about if the request processing has been successful.

In addition to the single pack transactions, the client has the possibility to enter the pack data manually. The difference between these transactions is that the manual transaction does not provide the batch data (*Single Pack Transaction - Request Format*).

The ‘undo’ transaction works the same as the ‘do’ transaction, except in case of an anonymous integration scenario (the different scenarios are described in Response Header).

The anonymous case is described in 9.2.2.

Preconditions:

The pharmacies and wholesalers must have a valid certificate to call the Single Pack Transactions.

10.1.1 Single Pack Transaction - Request Format

Attribute	O / M[1]	Description
B – Body Data		
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs.
Batch expiry date	M	Expiry date, equal for all packs belonging to the same batch. Enter date in format YYMMDD
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different.

[1] O = Optional Field / M = Mandatory Field

Attribute	O / M[1]	Description
		Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

Table 13: Body – single pack transaction request format

Example:

```
  <Body>
    <Product>
      <ProductCode scheme="GTIN">05060141900015</ProductCode>
      <Batch>
        <Id>1264</Id>
        <ExpDate>171212</ExpDate>
      </Batch>
    </Product>
    <Pack SN="XVTR75973491006155" />
  </Body>
</G110Request>
```

10.1.2 Single Pack Transaction - Request Format for Manual Data Entry

Attribute	O / M[2]	Description
B – Body Data		
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

[2] O = Optional Field / M = Mandatory Field

Table 14: Body – single pack transaction request format for manual data entry

Example:

```
<Body>
  <Product>
    <ProductCode scheme="GTIN">05060141900015</ProductCode>
  </Product>
  <Pack sn="XVTR75973491006155" />
</Body>
```

10.1.3 Single Pack Transaction – General Response Format

Attribute	O / M[3]	Description
B – Body Data		
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
Intermarket flag	O	Informs the caller that the requested pack data has not been found in the national database and that therefore an intermarket request has been launched.
NHRN	O	National healthcare reimbursement number, optional use in the pharmacy/wholesaler system
Extended master data	O	Reference to extended master data, e.g. leaflet
Product information date	O	Date of the release of this information
Product information changed	O	Indicated if this information was changed
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs.
Batch expiry date	O	Expiry date, equal for all packs belonging to the same batch.

[3] O = Optional Field / M = Mandatory Field

Attribute	O / M[3]	Description
NMVS batch expiry date	O	Shows the actual expiry date on NMVS which may be different due to batch update processes.
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.
Pack state	M	State of a pack
Pack state reasons	O	List of associated details or reasons why a pack is not able to be dispensed
B4 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 15: Body – Single Pack Transaction – General Response Format

Example:

```
<Body>
  <Product>
    <ProductCode scheme="GTIN">05060141900015</ProductCode>
    <!--Optional:-->
    <Intermarket>true</Intermarket>
    <!--Optional:-->
    <NHRN>09556886</NHRN>
    <!--Optional:-->
    <ProductInfoRef date="2017-09-02" modified="true">5765787852</ProductInfoRef>
    <!--Optional:-->
    <Batch>
      <Id>1264</Id>
      <!--Optional:-->
      <ExpDate>171231</ExpDate>
      <!--Optional:-->
      <ExpDateNMVS>2017-12-31</ExpDateNMVS>
    </Batch>
  </Product>
```

```
    <Pack SN="XVTR75973491006155" state="INACTIVE">
      <Reason>SUPPLIED</Reason>
    </Pack>
    <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
  </Body>
```

10.1.4 G110 - Verify Single Pack

This chapter describes the process of verifying a pack to integrate it into the EU supply chain. This process is applicable to a large number of different scenarios and actors. The use case describes the verification of a single pack by a distributor who connects to the national system. It is mostly used by pharmacists.

10.1.4.1 G110 - Process description

The respective pack is scanned by the pharmacist or wholesaler. The scanned data will be interpreted within the local software system to confirm if the pack code should contain serialized data. This internal check is to ensure that the scanned pack is in scope of the legislation and can be verified.

If the pack is in scope but the scanned code is not a serialized one or the code is a serialized one but not in scope a local information will be displayed to the user. There will be no general generated alerts.

There should be a local filtering in the respective IT system at the pharmacy/wholesaler to reduce the traffic as well as workload to the national system.

If the pack needs to be serialized and the code is a serialized one, the stakeholder system will submit the data to the national system where the product code and serial ID pair are used to look up the pack record. The batch code plus the expiry date scanned from the data matrix code are compared with the stored data in the national database.

Only if the product code, serial number, batch ID, and expiry date match, the pack record will be considered to be found. If the product is not known to the national system, the data will be forwarded to the European hub to check if this one is a ‘inter-market transaction’ process to identify if the scanned pack is known to another national system.

If the pack was found, an event record will be recorded for the verification event and the result of the verification request will be sent back to the respective IT-system at the pharmacy/wholesaler where the pack status and any associated pack information will be displayed.

If the pack is not found in the national system as well as when the ‘inter-market transaction’ request is negative, the result will be processed in compliant with the agreed alerts and exceptions, processing rules and a message will be sent to the local client that the pack was not found.

If an alert is appropriate according to the defined rules, an alert in the national system and/or an alert in the European Hub will be generated according to the alerts and exceptions processing rules.

10.1.4.2 G110 - Use Case Diagram

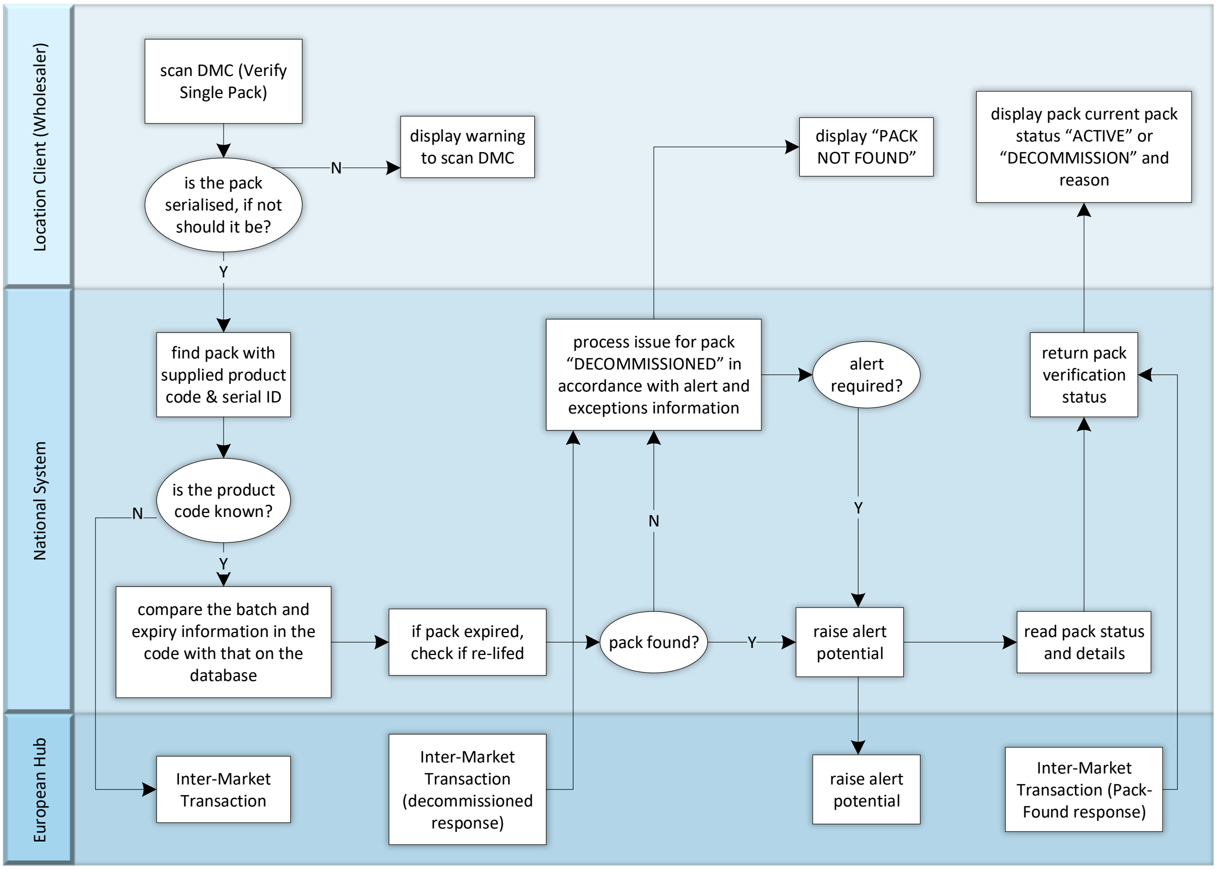

Location Client (Wholesaler)
National System
European Hub
scan DMC (Verify Single Pack)
is the pack serialised, if not should it be?
N
display warning to scan DMC
Y
display “PACK NOT FOUND”
display pack current pack status “ACTIVE” or “DECOMMISSION” and reason
find pack with supplied product code & serial ID
process issue for pack “DECOMMISSIONED” in accordance with alert and exceptions information
alert required?
return pack verification status
is the product code known?
N
Y
N
Y
compare the batch and expiry information in the code with that on the database
if pack expired, check if re-lifed
pack found?
Y
raise alert potential
read pack status and details
Inter-Market Transaction
Inter-Market Transaction (decommissioned response)
raise alert potential
Inter-Market Transaction (Pack-Found response)

Figure 20: Use Case Diagram Verify Single Pack

10.1.4.3 G110 - Web service description

To verify a single pack, the web service G110 needs to be used for further details. Please pay attention to “**Interface Description for Distributors.xls**” in the related interface version

10.1.5 G120 - Dispense Single Pack

This use case describes the dispensing of a single pack at a distributor who connects to the national system. It is a use case that is mostly used by pharmacists.

This use case may require additional functionality – as all the other use cases as well – to document the activities.

With the dispense pack use case the product pack gets decommissioned and the new state will be “Supplied”.

10.1.5.1 G120 - Process description

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The first step is to verify the pack. This process is described within the ‘Verify Pack’ use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive the following activities will continue. If the pack is not known, an alert following the rules will be generated. The pack status will be checked to see if the product has not already been dispensed before and if the status is 'Active'. In that case the pack status is set to 'Decommissioned/Supplied', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

If the pack has already been dispensed, the system will check if the same stakeholder has executed the dispensing. In that case the user accidently scanned a pack twice – a so called 'double dispensing' operation. A specific counter for double dispensing is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double dispensing.

If the distributor is not the same or the count of double dispensing is exceeding the agreed limits, an alert following the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.5.2 G120 - Use Case Diagram

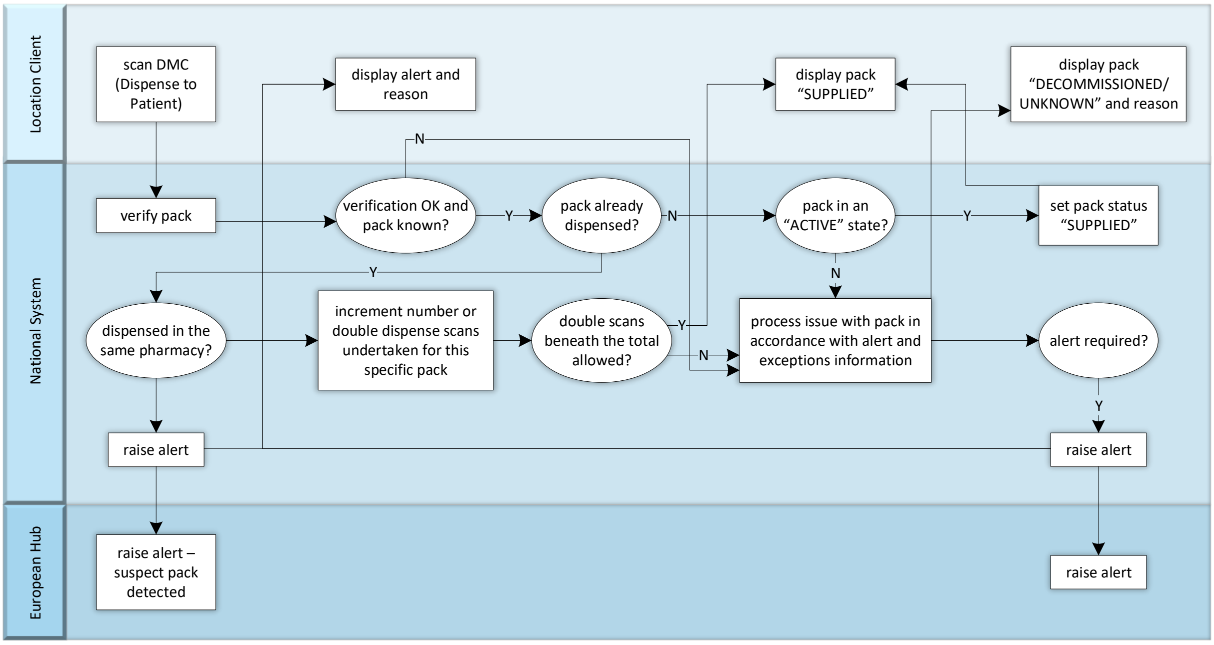

Location Client
National System
European Hub
scan DMC (Dispense to Patient)
display alert and reason
display pack "SUPPLIED"
display pack "DECOMMISSIONED/ UNKNOWN" and reason
verify pack
verification OK and pack known?
pack already dispensed?
pack in an "ACTIVE" state?
set pack status "SUPPLIED"
dispensed in the same pharmacy?
increment number or double dispense scans undertaken for this specific pack
double scans beneath the total allowed?
process issue with pack in accordance with alert and exceptions information
alert required?
raise alert
raise alert
raise alert – suspect pack detected
raise alert
Y
N
Y
N
Y
N
Y
N
Y

Figure 21: Web Service Dispense Single Pack

10.1.5.3 G120 - Web Service Description

The web service G120 is used to dispense a single pack. Please pay attention to document Interface Description for Distributors.xls in the related interface version for further details.

10.1.6 G122 - Dispense Single Pack – Manual Data Entry

This use case describes the dispensing of a single pack by a distributor who is connected to the national system. It is a use case that is mostly used by pharmacists.

This use case is serving as a fall back procedure when there is a problem with the distributor's IT system so that the user is not able to scan the pack due to a problem with the scanning device. For that case the user can manually type in the pack data.

With the dispense pack use case the product pack gets decommissioned and the new pack state becomes "Supplied".

10.1.6.1 G122 - Process Description

The data of the pack in hand is manually entered into the used IT system. To minimize the risk of typos only the serial number and the product code need to be entered. A first check-up will be performed by the distributor's IT system. If for instance there is a problem with the check sum of the GTIN, it will be necessary to inform the operator by the local system. If the pre-check detects no errors, the manually entered data will be transferred to the national system. The first step is to verify the pack. This process is described within the 'Verify Pack' use case (see chapter 10.1). The only difference is to make the verification without batch ID and expiry date.

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert according to the rules will be generated. The pack status is checked if the product has not already been dispensed before and if the status is 'Active'. In that case, the pack status is set to 'Decommissioned/Supplied', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator. If the status is already decommissioned, an alert following the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

If the outcome of the verification is that the pack is not known the assumption will be that the operator executed a typo by entering the pack data. A specific counter for typos is incremented. While the counter is within the agreed limits, a feedback is submitted to the distributor's IT system to be displayed to the operator to type in again the requested information. The process will start again at the beginning. If the count of typos is exceeding the agreed limits, an alert according the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Additional alerts following the appropriate alert and exception rules will be generated within the national System.

If the pack has already been dispensed, the system will check if the same stakeholder has executed the previous dispensing. In that case the user accidently entered the pack data twice – a so called 'double dispensing' operation. A specific counter for double dispensing is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double dispensing.

If the stakeholder is not the same or the count of double dispensing exceeded the agreed limits, an alert according to the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Additional alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.6.2 G122 - Use Case Diagram

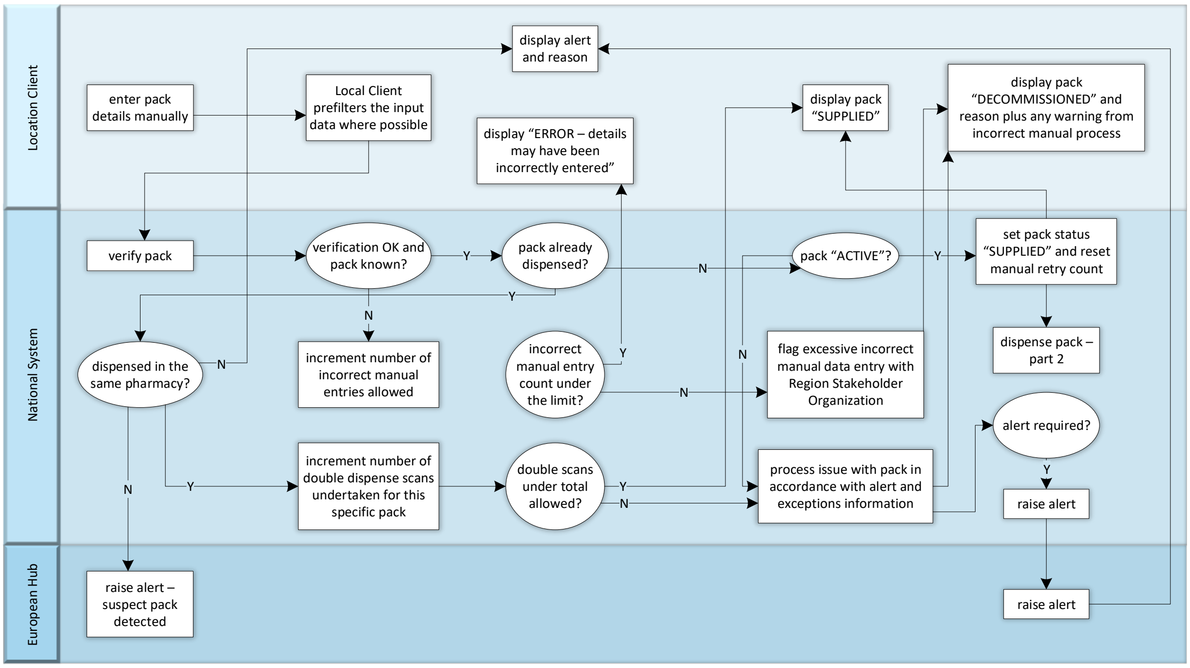

Location Client
National System
European Hub
enter pack details manually
Local Client prefilters the input data where possible
display alert and reason
display "ERROR – details may have been incorrectly entered"
display pack "SUPPLIED"
display pack "DECOMMISSIONED" and reason plus any warning from incorrect manual process
verify pack
verification OK and pack known?
Y
pack already dispensed?
N
pack "ACTIVE"?
Y
set pack status "SUPPLIED" and reset manual retry count
Y
N
dispensed in the same pharmacy?
N
increment number of incorrect manual entries allowed
incorrect manual entry count under the limit?
Y
N
flag excessive incorrect manual data entry with Region Stakeholder Organization
N
dispense pack – part 2
alert required?
Y
N
Y
increment number of double dispense scans undertaken for this specific pack
double scans under total allowed?
Y
N
process issue with pack in accordance with alert and exceptions information
raise alert
raise alert – suspect pack detected
raise alert

Figure 22: Dispense Single Pack – Manual Data Entry

10.1.6.3 G122 - Web Service Description

The web service G122 needs to be used to dispense a single pack with manual data entry at the dispensing point.

10.1.7 G130 - Destroy Single Pack

This use case describes the procedure when a stakeholder wants to mark a single pack as destroyed. The pack will get the status 'Destroyed'.

10.1.7.1 G130 - Process Description

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The first step is to verify the pack. This process is described within the 'Verify Pack' use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert according to the rules will be generated. The pack status is checked if the product has not already been dispensed before and if the status is 'Active' or 'Locked', in that case, the pack status is set to 'Destroyed', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

If the pack has already been decommissioned, the system will check if the same stakeholder has executed the decommissioning. In that case, the user accidently scanned a pack twice – a so called 'double decommissioning' operation. A specific counter for double decommissioning is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double decommissioning.

If the stakeholder is not the same, or the count of double decommissioning exceeded the agreed limits, an alert according to the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.7.2 G130 - Use Case Diagram

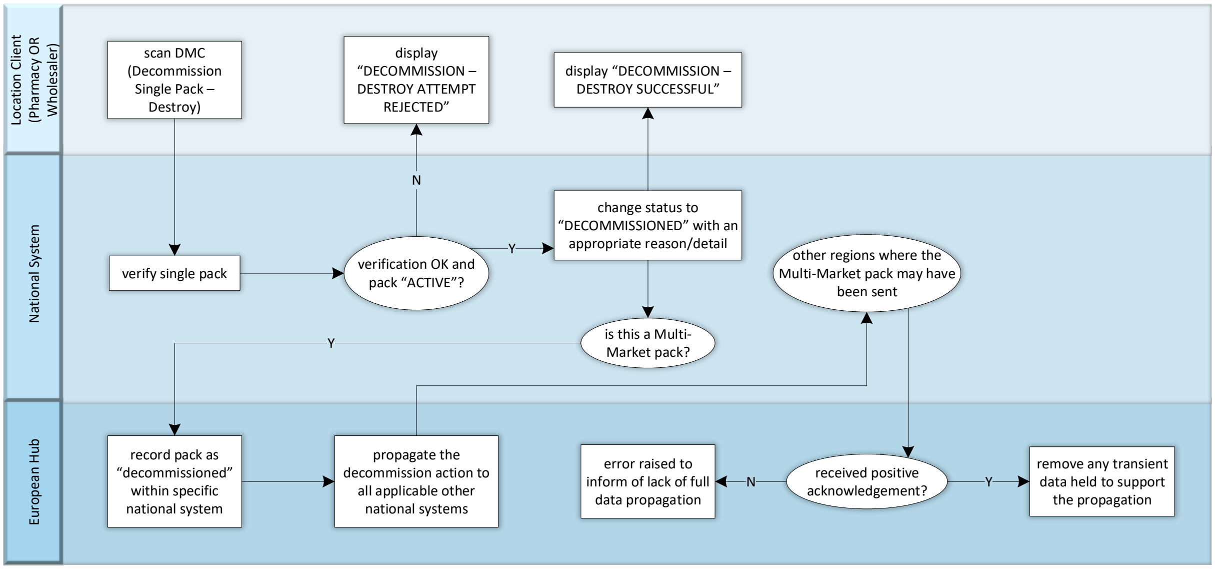

Location Client (Pharmacy OR Wholesaler)
National System
European Hub
scan DMC (Decommission Single Pack – Destroy)
display "DECOMMISSION – DESTROY ATTEMPT REJECTED"
display "DECOMMISSION – DESTROY SUCCESSFUL"
N
verify single pack
verification OK and pack "ACTIVE"?
Y
change status to "DECOMMISSIONED" with an appropriate reason/detail
other regions where the Multi-Market pack may have been sent
is this a Multi-Market pack?
Y
record pack as "decommissioned" within specific national system
propagate the decommission action to all applicable other national systems
error raised to inform of lack of full data propagation
N
received positive acknowledgement?
Y
remove any transient data held to support the propagation

Figure 23: Destroy Single Pack

10.1.8 G140 - Export Single Pack

This scenario describes the case when a wholesaler wants to export a pack with EU safety feature. The pack will be reported to the national system as 'Exported'.

10.1.8.1 G140 - Process Description

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The first step is to verify the pack. This process is described within the 'Verify Pack' use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert according to the rules will be generated. The pack status is checked if the product has not already been dispensed before and if the status is 'Active'. In that case, the pack status is set to 'Exported', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

If the pack has already been decommissioned, the system will check if the same stakeholder has executed the decommissioning. In that case, the user accidently scanned a pack twice – a so called 'double decommissioning' operation. A specific counter for double decommissioning is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double decommissioning.

If the stakeholder is not the same or the count of double decommissioning exceeds the agreed limits, an alert according to the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.8.2 G140 - Use Case Diagram

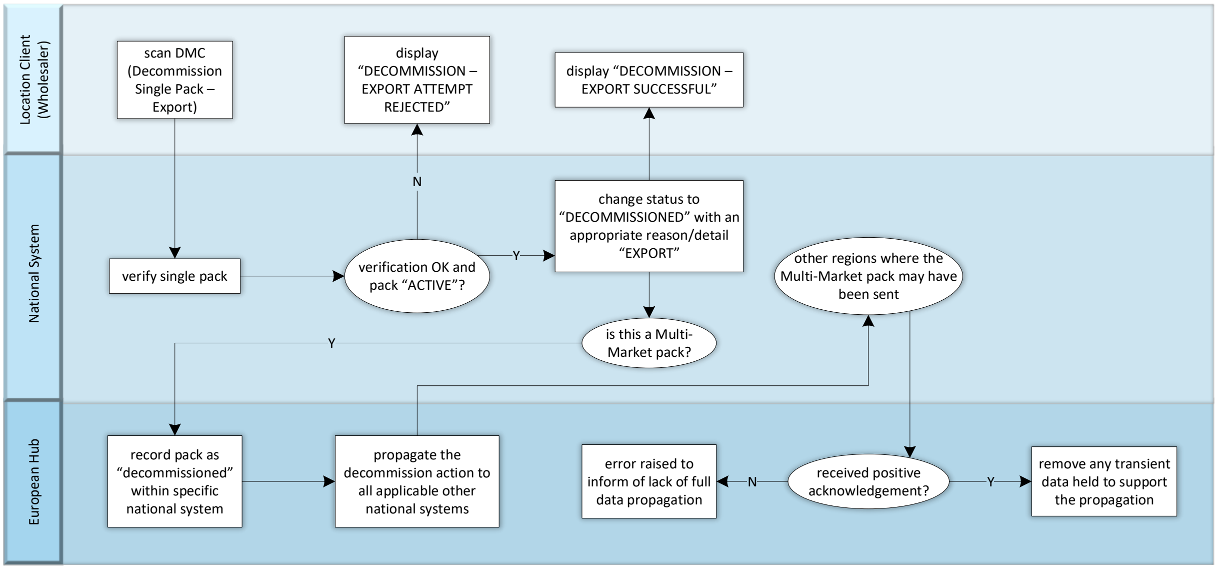

Location Client (Wholesaler)
National System
European Hub
scan DMC (Decommission Single Pack – Export)
display "DECOMMISSION – EXPORT ATTEMPT REJECTED"
display "DECOMMISSION – EXPORT SUCCESSFUL"
N
change status to "DECOMMISSIONED" with an appropriate reason/detail "EXPORT"
verify single pack
verification OK and pack "ACTIVE"?
Y
other regions where the Multi-Market pack may have been sent
Y
is this a Multi-Market pack?
record pack as "decommissioned" within specific national system
propagate the decommission action to all applicable other national systems
error raised to inform of lack of full data propagation
N
received positive acknowledgement?
Y
remove any transient data held to support the propagation

Figure 24: Export Single Pack

10.1.9 G150 – Sample Single Pack

Packs marked as 'Sample' are packs provided to national authorities, clinical trials and investigations. The use case is the same like the one that already has been described in chapter 10.1.5 G120 - Dispense Single Pack. The only difference is the status at the end of the process which will be 'Sample'.

10.1.9.1 G150 - Process Description

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The first step is to verify the pack. This process is described within the 'Verify Pack' use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert according to the rules will be generated. The pack status is checked if the product has not been dispensed before and if the status is 'Active'. In that case, the pack status is set to 'Sample', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

If the pack has already been decommissioned, the system will check if the same stakeholder has executed the decommissioning. In that case, the user accidently scanned a pack twice – a so called 'double decommissioning' operation. A specific counter for double decommissioning is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double decommissioning.

If the stakeholder is not the same or the count of double decommissioning is exceeding the agreed limits, an alert following the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.9.2 G150 - Use Case Diagram

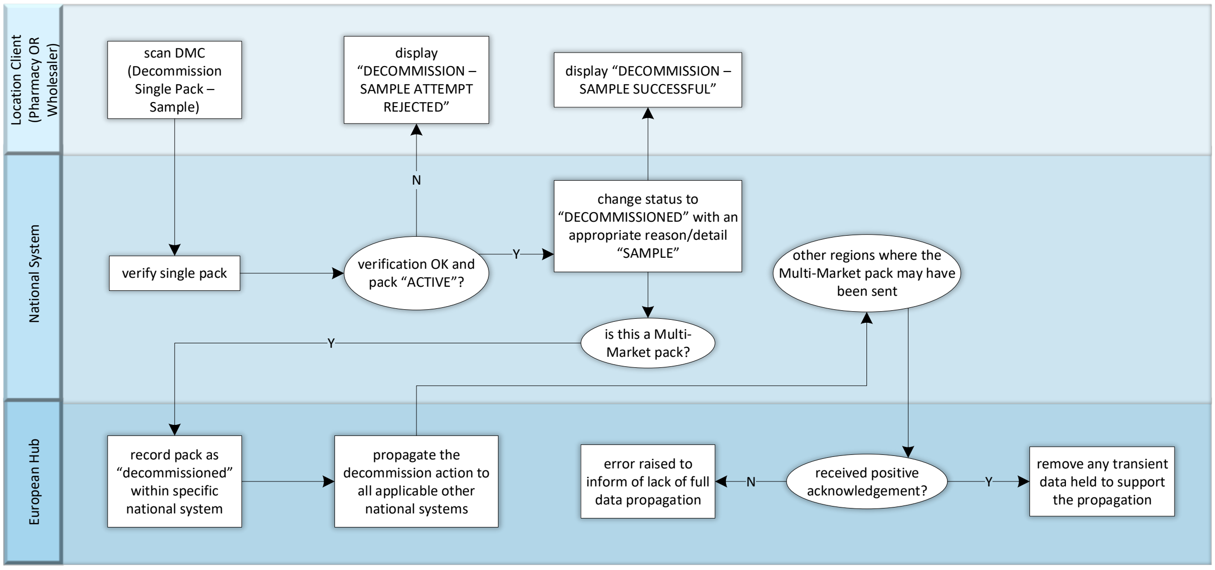

Location Client (Pharmacy OR Wholesaler)
National System
European Hub
scan DMC (Decommission Single Pack – Sample)
display "DECOMMISSION – SAMPLE ATTEMPT REJECTED"
display "DECOMMISSION – SAMPLE SUCCESSFUL"
N
change status to "DECOMMISSIONED" with an appropriate reason/detail "SAMPLE"
verify single pack
verification OK and pack "ACTIVE"?
Y
other regions where the Multi-Market pack may have been sent
Y
is this a Multi-Market pack?
record pack as "decommissioned" within specific national system
propagate the decommission action to all applicable other national systems
error raised to inform of lack of full data propagation
N
received positive acknowledgement?
Y
remove any transient data held to support the propagation

Figure 25: Supply Single Pack as Sample (e.g. for Authorities)

10.1.10 G160 – Free Sample Single Pack

This use case describes the procedure when a stakeholder wants to set the status of a single pack to a free sample. The pack will get the new status 'Free Sample'.

10.1.10.1 G160 - Process Description

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The first step is to verify the pack. This process is described within the 'Verify Pack' use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert according to the rules will be generated. The pack status is checked if the product has not already been dispensed before and if the status is 'Active' or 'Locked'. In that case, the pack status is set to 'Free Sample', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

If the pack has already been decommissioned, the system will check if the same stakeholder has executed the decommissioning. In that case, the user accidently scanned a pack twice – a so called 'double decommissioning' operation. A specific counter for double decommissioning is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double decommissioning.

If the stakeholder is not the same or the count of double decommissioning exceeds the agreed limits, an alert according to the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.10.2 G160 - Use Case Diagram

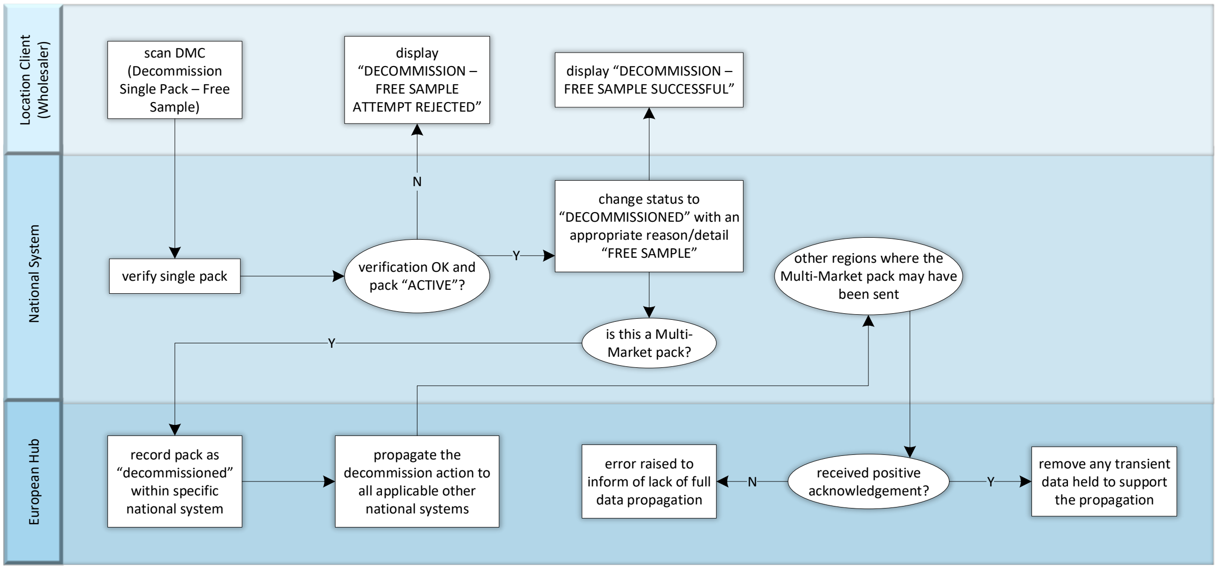

Location Client (Wholesaler)
National System
European Hub
scan DMC (Decommission Single Pack – Free Sample)
display "DECOMMISSION – FREE SAMPLE ATTEMPT REJECTED"
display "DECOMMISSION – FREE SAMPLE SUCCESSFUL"
N
change status to "DECOMMISSIONED" with an appropriate reason/detail "FREE SAMPLE"
Y
verify single pack
verification OK and pack "ACTIVE"?
other regions where the Multi-Market pack may have been sent
Y
is this a Multi-Market pack?
record pack as "decommissioned" within specific national system
propagate the decommission action to all applicable other national systems
error raised to inform of lack of full data propagation
N
received positive acknowledgement?
Y
remove any transient data held to support the propagation

Figure 26: Sample Single Pack

10.1.11 G170 - Lock Single Pack

The Lock Pack use case can be used to set a pack status for quarantining a pack. The new state of the pack is "Locked".

10.1.11.1 G170 - Process Description

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The first step is to verify the pack. This process is described within the 'Verify Pack' use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert according to the rules will be generated. The pack status is checked if the product has not already been dispensed before and if the status is 'Active'. In that case, the pack status is set to 'Locked', a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

If the pack has already been decommissioned, the system will check if the same stakeholder has executed the decommissioning. In that case, the user accidently scanned a pack twice – a so called 'double decommissioning' operation. A specific counter for double decommissioning is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor's IT system to be displayed to the operator – with a comment about double decommissioning.

If the stakeholder is not the same or the count of double decommissioning exceeds the agreed limits, an alert following the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.11.2 G170 - Use Case Diagram

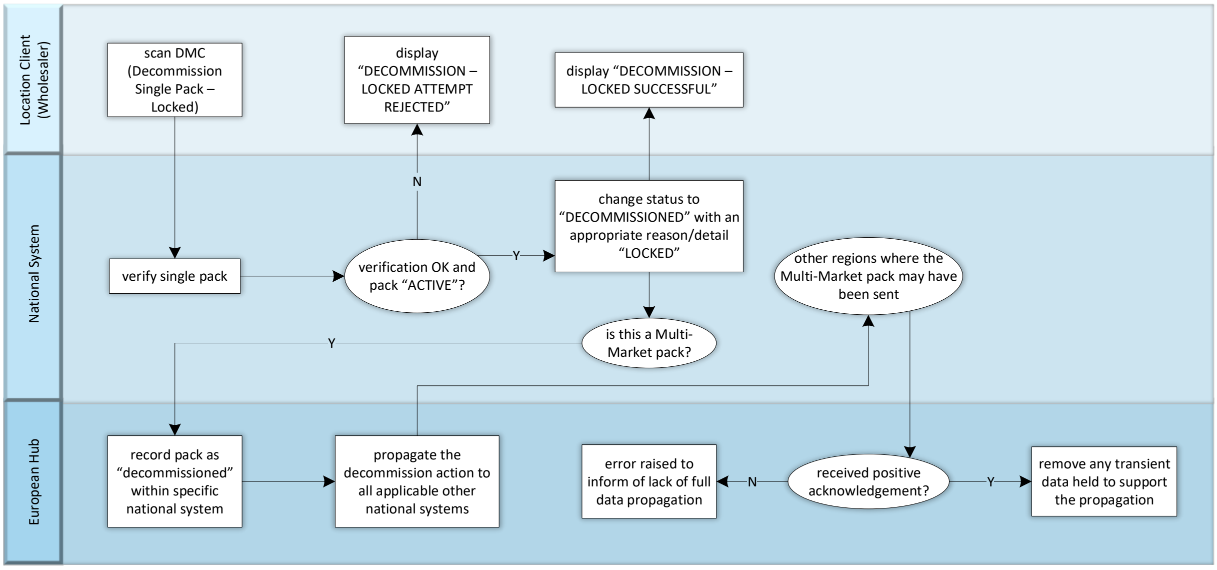

Location Client (Wholesaler)
National System
European Hub
scan DMC (Decommission Single Pack – Locked)
display “DECOMMISSION – LOCKED ATTEMPT REJECTED”
display “DECOMMISSION – LOCKED SUCCESSFUL”
N
change status to “DECOMMISSIONED” with an appropriate reason/detail “LOCKED”
Y
verify single pack
verification OK and pack “ACTIVE”?
other regions where the Multi-Market pack may have been sent
is this a Multi-Market pack?
Y
record pack as “decommissioned” within specific national system
propagate the decommission action to all applicable other national systems
error raised to inform of lack of full data propagation
N
received positive acknowledgement?
Y
remove any transient data held to support the propagation

Figure 27: Lock Single Pack

10.1.12 G180 - Stolen Single Pack

This use case describes the procedure when a stakeholder wants to mark a single pack as stolen. The pack will get the new status ‘Stolen’.

10.1.12.1 G180 - Process Description

If by any chance the codes of a stolen pack were previously photocopied or they exist in any other form that can be scanned, even though the pack itself is not at hand anymore, the G180 use case can be used to mark the pack as stolen. Otherwise, the pack in hand cannot be scanned by the operator of the used IT system anymore since the pack was stolen and is not present. In the latter case the serial number and additional data have to be looked up by the operator in his merchandise management system and manually typed in using the web interface. Then the data is submitted to the national system.

The first step is to verify the pack. This process is described within the ‘Verify Pack’ use case (see chapter 10.1).

Only if the verification check including the inter-market transaction is positive, the following activities will continue. If the pack is not known, an alert following the rules will be generated. The pack status is checked if the product has not already been dispensed before and if the status is ‘Active’ or ‘Locked’. In that case the pack status is set to ‘Stolen’, a protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor’s IT system to be displayed to the operator.

If the pack has already been decommissioned, the system will check if the same stakeholder has executed the decommissioning. In that case, the user accidently scanned a pack twice – a so called ‘double decommissioning’ operation. A specific counter for double decommissioning is incremented. While the counter is within the agreed limits, a positive feedback is submitted to the distributor’s IT system to be displayed to the operator – with a comment about double decommissioning.

If the stakeholder is not the same or the count of double decommissioning exceeds the agreed limits, an alert following the rules will be generated and a corresponding message will be sent to the distributor's IT system to be displayed to the operator. Alerts following the appropriate alert and exception rules will be generated within the national and / or European System.

10.1.12.2 G180 - Use Case Diagram

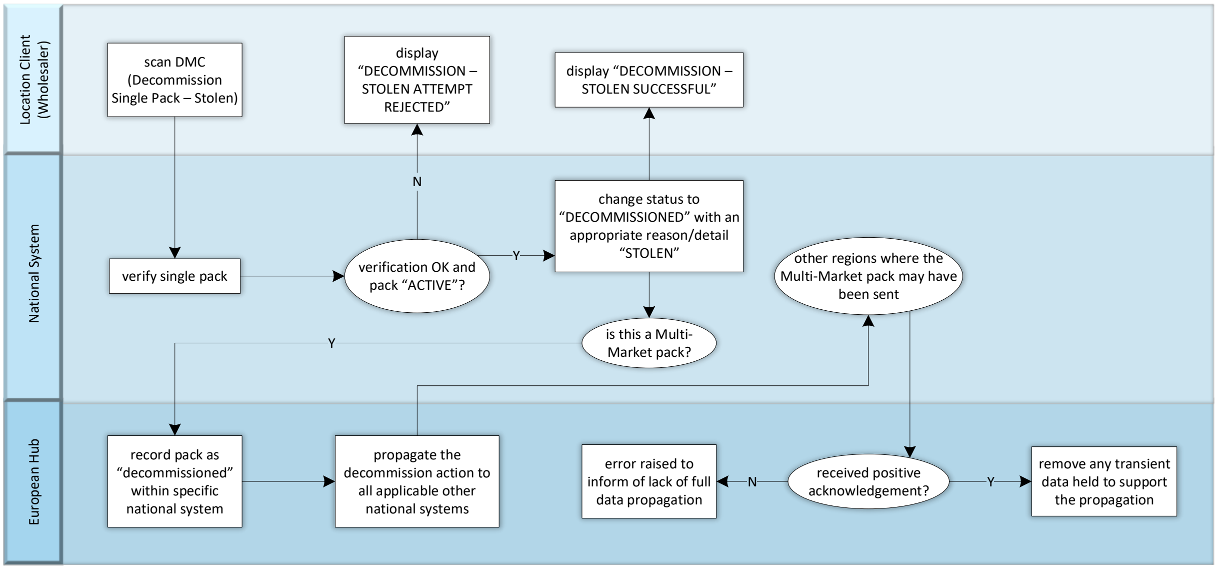

Location Client (Wholesaler)
National System
European Hub
scan DMC (Decommission Single Pack – Stolen)
display "DECOMMISSION – STOLEN ATTEMPT REJECTED"
display "DECOMMISSION – STOLEN SUCCESSFUL"
N
verify single pack
verification OK and pack "ACTIVE"?
Y
change status to "DECOMMISSIONED" with an appropriate reason/detail "STOLEN"
other regions where the Multi-Market pack may have been sent
is this a Multi-Market pack?
Y
record pack as "decommissioned" within specific national system
propagate the decommission action to all applicable other national systems
error raised to inform of lack of full data propagation
N
received positive acknowledgement?
Y
remove any transient data held to support the propagation

Figure 28: Stolen Single Pack

10.2 Single Pack Transactions - Undo Decommission

Each single pack service that changes the state of one pack has an 'Undo' service which reverts the changed status.

Only with the anonymous integration scenario a unique reference ID to the previous transaction is necessary to identify the initial request/requestor. With the other integration scenarios, the system will verify the client IDs of the requestor and the period of time between the 'Undo' transaction and the initial single transaction. Only the initial requestor is able to undo the changed status.

Preconditions:

The pharmacies and wholesalers must have a valid certificate to call the 'Single Pack Transactions'.

In the anonymous integration scenario:
In addition to the check of the ClientLoginID and UserID, the RefClientTrxID of the original transaction – the one to be reversed – is needed to perform the 'Undo' transaction.

The 'Undo' processes of removing packs out of the supply chain are described in the following sub chapters. The functionality is similar to the decommission procedure. The 'Undo' procedures are clearly defined in the delegated acts.

Only the distributor who accomplished the decommission procedure is able to trigger off the 'Undo' procedure. It must be the same member, user ID and sub-user ID (if a stakeholder-member-system is in place between NMVS and pharmacy/wholesaler system). If the user could not be uniquely identified by the set of IDs, the original transaction ID of the ‘do’ service that should be undone will need to be submitted.

The ‘Undo’ procedure must be triggered off within an agreed time frame (max 10 days after the decommission process).

10.2.1 Undo Single Pack Transaction – Request Format

Attribute	O / M[4]	Description
B – Body Data		
Reference client transaction ID	O	Unique identifier of the transaction from the client system
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs.
Batch expiry date	M	Expiry date, equal for all packs belonging to the same batch. Date in format YYMMDD
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

Table 16: Body – Undo Single Pack Transaction Request Format

Example:

```
<Body>
  <Product>
    <ProductCode scheme="GTIN">05060141900015</ProductCode>
```

[4] O = Optional Field / M = Mandatory Field

```
    <Batch>
      <Id>1264</Id>
      <ExpDate>171200</ExpDate>
    </Batch>
  </Product>
  <Pack sn="XVTR75973491006155" />
  <!--Optional:-->
  <RefClientTrxId>c295a450-0caa-4cf1-8e62-c6165c3</RefClientTrxId>
</Body>
```

10.2.2 G121 - Undo Dispense Single Pack

This use case describes the reversal of the dispensing of a single pack. It is used by pharmacists and wholesalers.

10.2.2.1 G121 - Process Description

In this case, a distributor wants to re-introduce an already dispensed pack back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the dispensing process of this specific pack must not be more than 10 days (240h) and the current status must be ‘Dispensed’. In that case, the national system will check if there is any other given status in the meanwhile with this specific pack like recalled, withdrawn, or expired. If this is not the case, the national system will make the pack ‘Active’ again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor’s IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor’s IT system to be displayed to the operator with the respective reason.

10.2.2.2 G121 - Use Case Diagram

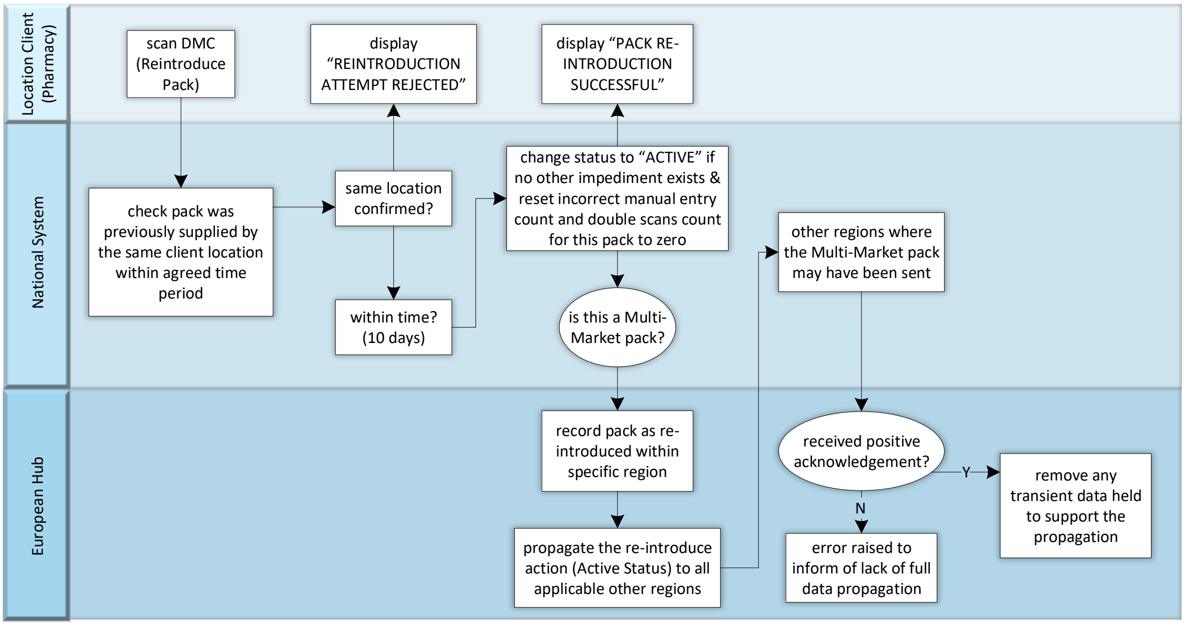

Location Client (Pharmacy)
National System
European Hub
scan DMC (Reintroduce Pack)
display "REINTRODUCTION ATTEMPT REJECTED"
display "PACK RE-INTRODUCTION SUCCESSFUL"
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
change status to "ACTIVE" if no other impediment exists & reset incorrect manual entry count and double scans count for this pack to zero
is this a Multi-Market pack?
other regions where the Multi-Market pack may have been sent
record pack as re-introduced within specific region
propagate the re-introduce action (Active Status) to all applicable other regions
received positive acknowledgement?
Y
N
remove any transient data held to support the propagation
error raised to inform of full data propagation

Figure 29: Undo Dispense Single Pack

10.2.2.3 G121 - Web Service Description

The web service G121 needs to be used to undo dispense a single pack. For further details, please pay attention to "Interface Description for Distributors.xls" in the related interface version

10.2.3 G123 - Undo Dispense Single Pack – Manual Data Entry

Functional equivalent process to G121 – with the difference, that the user nether needs the batch no. nor the expiry date. Just the product code and the serial number is required.

This use case is a fallback procedure in case the pack is no longer available but the data is known anyhow or the scanning system fails. For that case the user can manually type in the pack data.

10.2.3.1 G123 - Web Service Description

The web service G123 needs to be used to undo dispense a single pack For further details please pay attention to Interface Description for Distributors.xls in the related interface version

10.2.3.2 G131 - Process Description

In this case a distributor wants to re-introduce a pack which is marked as destroyed back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the destroyed process of this specific pack must not be more than 10 days (240h) and the current status must be 'Destroyed'. In that case, the national system will check if there is any other given status in the meanwhile with this specific pack like 'Recalled', 'Withdrawn', or 'Expired'. If this is not the case, the national system will make the pack 'Active' again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor's IT system to be displayed to the operator with the respective reason.

10.2.3.3 G131 - Use Case Diagram

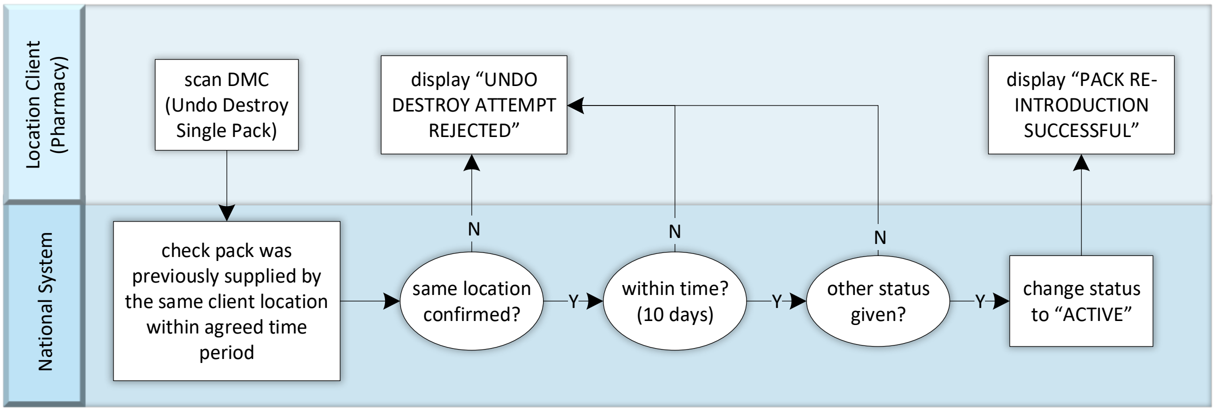

Location Client (Pharmacy)
National System
scan DMC (Undo Destroy Single Pack)
display "UNDO DESTROY ATTEMPT REJECTED"
display "PACK RE-INTRODUCTION SUCCESSFUL"
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
other status given?
change status to "ACTIVE"
N
N
N
Y
Y
Y

Figure 30: Undo Destroy Single Pack

10.2.4 G141 - Undo Export Single Pack

This use case describes the reversal of an export of a single pack. It is used by wholesalers.

10.2.4.1 G141 - Process Description

In this case, a wholesaler wants to re-introduce a pack which is marked as exported back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the export process of this specific pack must not be more than 10 days (240h) and the current status must be 'Exported'. In that case, the national system will check if there is any other given status in the meanwhile with this specific pack like 'Recalled', 'Withdrawn', or 'Expired'. If this is not the case, the national system will make the pack 'Active' again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor's IT system to be displayed to the operator with the respective reason.

10.2.4.2 G141 - Use Case Diagram

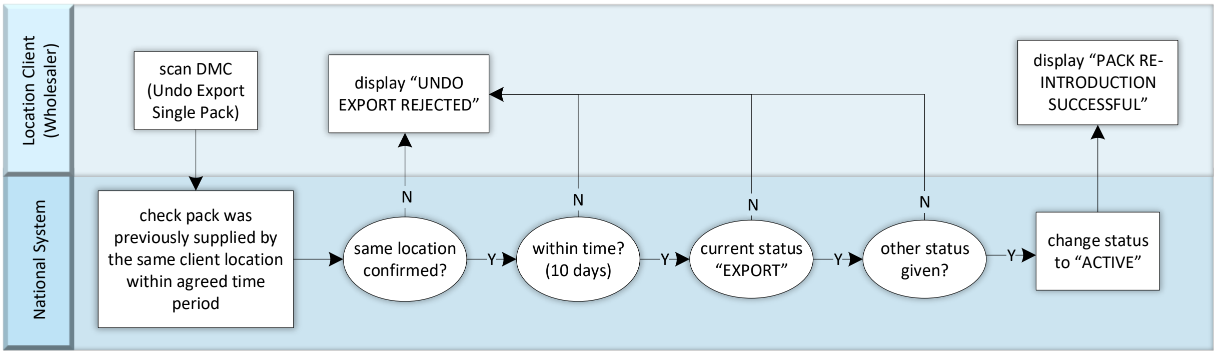

Location Client (Wholesaler)
National System
scan DMC (Undo Export Single Pack)
display "UNDO EXPORT REJECTED"
display "PACK RE-INTRODUCTION SUCCESSFUL"
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
current status "EXPORT"
other status given?
change status to "ACTIVE"
N
N
N
N
Y
Y
Y
Y

Figure 31: Undo Export Single Pack

10.2.5 G151 - Undo Sample Single Pack

This use case describes the reversal of the sample of a single pack. It is used by pharmacists and wholesalers.

10.2.5.1 G151 - Process Description

In this case a pharmacist, wholesaler or dispending doctor wants to re-introduce a pack which is marked as a sample back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the sample process of this specific pack must not be more than 10 days (240h) and the current status must be 'Sample'. In that case, the national system will check if there is any other given status in the meanwhile with this specific pack like 'Recalled', 'Withdrawn', or 'Expired'. If this is not the case, the national system will make the pack 'Active' again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor's IT system to be displayed to the operator with the respective reason.

10.2.5.2 G151 - Use Case Diagram

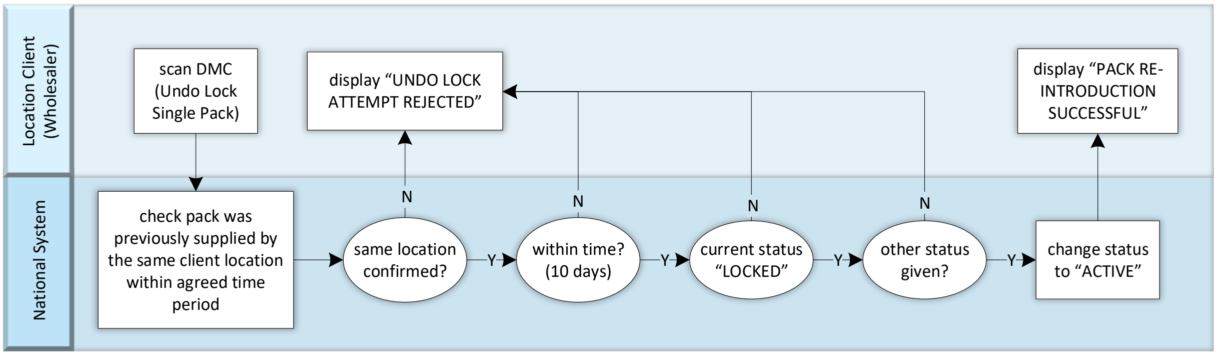

Location Client (Wholesaler)
National System
scan DMC (Undo Lock Single Pack)
display "UNDO LOCK ATTEMPT REJECTED"
display "PACK RE-INTRODUCTION SUCCESSFUL"
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
current status "LOCKED"
other status given?
change status to "ACTIVE"
N
N
N
N
Y
Y
Y
Y

Figure 32: Undo Sample Single Pack

10.2.6 G161 - Undo Free Sample Single Pack

This use case describes the reversal of the free-sample of a single pack. It is used by a wholesaler.

10.2.6.1 G161 - Process Description

In this case a wholesaler wants to re-introduce a pack which is marked as a free sample back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the free-sample process of this specific pack must not be more than 10 days (240h) and the current status must be ‘Free-Sample’. In that case, the national system will check if there is any other given status in the meanwhile with this specific pack like ‘Recalled’, ‘Withdrawn’, or ‘Expired’. If this is not the case, the national system will make the pack ‘Active’ again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor’s IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor’s IT system to be displayed to the operator with the respective reason.

10.2.6.2 G161 - Use Case Diagram

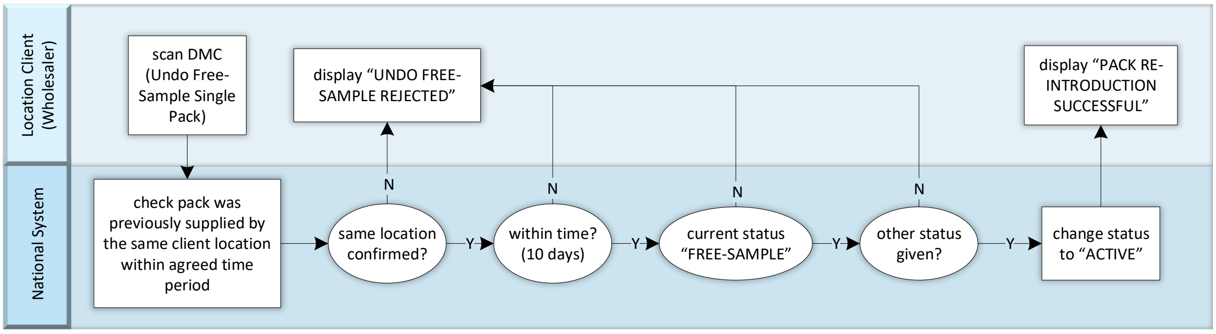

Location Client (Wholesaler)
National System
scan DMC (Undo Free-Sample Single Pack)
display “UNDO FREE-SAMPLE REJECTED”
display “PACK RE-INTRODUCTION SUCCESSFUL”
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
current status “FREE-SAMPLE”
other status given?
change status to “ACTIVE”
N
N
N
N
Y
Y
Y
Y

Figure 33: Undo FreeSample Single Pack

10.2.7 G171 - Undo Lock Single Pack

This use case describes the reversal of the lock of a single pack. It is used by a wholesaler.

10.2.7.1 G171 - Process Description

In this case a wholesaler wants to re-introduce a pack which is marked as locked back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the locked process of this specific pack must not be more than 10 days (240h) and the current status must be ‘Locked’. In that case the national system will check if there is any other given status in the meanwhile with this specific pack like ‘Recalled’, ‘Withdrawn’, or ‘Expired’. If this is not the case, the national system will make the pack ‘Active’ again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor’s IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor's IT system to be displayed to the operator with the respective reason.

10.2.7.2 G171 - Use Case Diagram

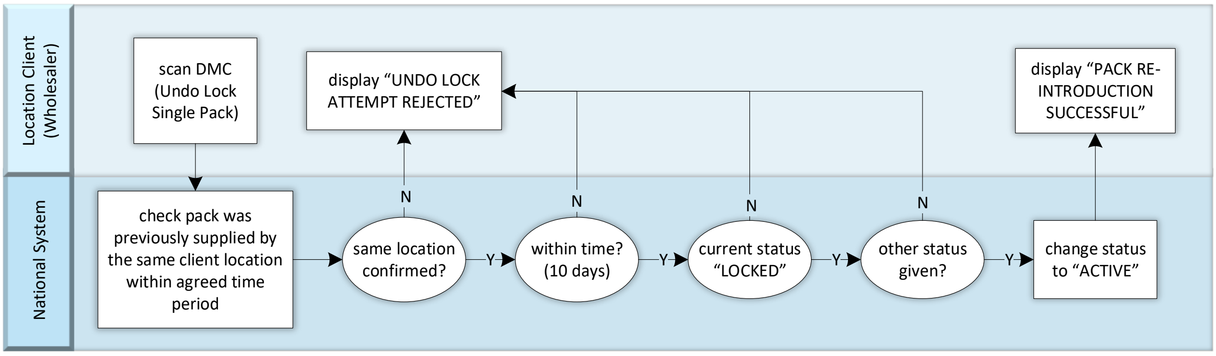

Location Client (Wholesaler)
National System
scan DMC (Undo Lock Single Pack)
display "UNDO LOCK ATTEMPT REJECTED"
display "PACK RE-INTRODUCTION SUCCESSFUL"
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
current status "LOCKED"
other status given?
change status to "ACTIVE"
N
N
N
N
Y
Y
Y
Y

Figure 34: Undo Lock Single Pack

10.2.8 G181 - Undo Stolen Single Pack

This function is no longer available since interface V2.

This use case describes the reversal of the marked as stolen of a single pack. It is used by pharmacists and wholesalers.

Important: Despite of the fact that this transaction is implemented in the system and described below for the sake of completeness, the URS does not foresee the transaction to be used. For that purpose the usage of that transaction is prevented by the rights and roles management of the NMVS.

In the future, if the URS will change and the usage of that particular transaction will become permitted, the transaction will be unlocked.

Currently, please do not implement the transaction in your client software. You will not be able to use it, anyway.

10.2.8.1 G181 - Process Description

In this case a distributor wants to re-introduce a pack which is marked as stolen back into the EU environment.

The pack in hand is scanned by the operator of the used IT system and submitted to the national system. The national system will check if it was scanned at the same location (not necessarily by the same operator). The period of time since the stolen process of this specific pack must not be more than 10 days (240h) and the current status must be 'Stolen'. In that case, the national system will check if there is any other given status in the meanwhile with this specific pack like 'Recalled', 'Withdrawn', or 'Expired'. If this is not the case, the national system will make the pack 'Active' again. A protocol of this process is recorded within the national system and the positive feedback is submitted to the distributor's IT system to be displayed to the operator.

In any other case, the undo request will be rejected. A protocol of this rejection will be recorded and a rejection feedback will be submitted to the distributor's IT system to be displayed to the operator with the respective reason.

10.2.8.2 G181 - Use Case Diagram

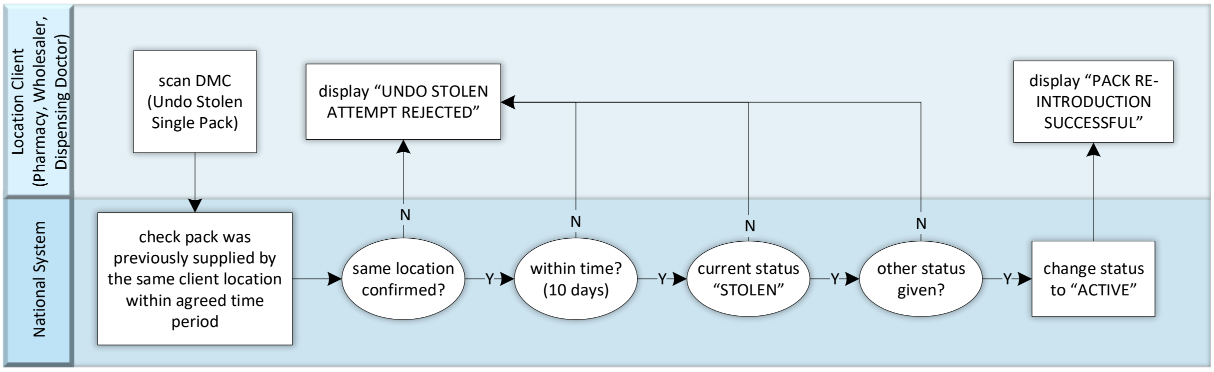

Location Client (Pharmacy, Wholesaler, Dispensing Doctor)
National System
scan DMC (Undo Stolen Single Pack)
display “UNDO STOLEN ATTEMPT REJECTED”
display “PACK RE-INTRODUCTION SUCCESSFUL”
check pack was previously supplied by the same client location within agreed time period
same location confirmed?
within time? (10 days)
current status “STOLEN”
other status given?
change status to “ACTIVE”
N
N
N
N
Y
Y
Y
Y

Figure 35: Undo Stolen Single Pack

11 Asynchronous Web Service Operations – Homogenous Bulk Transaction

11.1 Process Description

Bulk transaction describes the service to be used by a distributor to handle a bigger number of packs with the same product code. This use case is used by wholesalers and pharmacies. Unlike the Single Pack Transaction, the distributor’s IT system collects the serial numbers of each scanned pack within an internal buffer. After finalization of the scan process, the system will send out one transaction with the product code and the entire set of serial numbers.

As the Bulk Transaction is an asynchronous workflow the output of the Bulk Transaction is only the confirmation that the national system correctly received all information. The wholesaler or pharmacy system needs to re-ask for detailed information after a while. Depending on the number of scanned packs it can take some time to perform the Bulk Transaction within the national system. The minimum time to send out the request for the processing results should be at least more than 1 minute. The request needs to contain the original NMVS transaction ID from the output of the Bulk Transaction.

The homogenous bulk transactions operate on more than one single pack and carry out the same state change for all of the provided packs. All provided packs need to be part of a set of packs and need to belong to the same product type and product batch.

The national system receives the provided pack data and triggers off the processing of each provided pack. The response to the initial transaction is only a confirmation for the incoming request. The result of the processed pack data is passed back asynchronously and will be described in section 10 of this document.

The ‘Undo’ transaction works the same way like the ‘Do’ transaction, except in case of an anonymous integration scenario (the different scenarios are described in chapter 8.4.1.2 - Response Header).

Preconditions:

The asynchronous use cases cannot be processed in parallel because of the strict order of the incoming requests. This strict observance of order is important to avoid processing errors and general malfunction of the system.

Like the single pack transaction, the users need to have a valid certificate to call the bulk transactions.

11.2 Homogenous Bulk Transaction Use Cases

The following table shows the different use cases with the respective use case numbers. As the bulk services are asynchronous services there is a dedicated web service per use case to get the result.

Use Case Numbers	Business-Process
Homogenous Bulk Transactions – Do-Transactions	
G115	Bulk Verify Packs
G125	Bulk Dispense Packs
G135	Bulk Destroy Packs

Use Case Numbers	Business-Process
G145	Bulk export packs to Non-EU
G155	Bulk sample packs
G165	Bulk free sample packs
G175	Bulk lock packs
G185	Bulk stolen packs
Homogenous Bulk Transactions – Undo-Transactions	
G127	Bulk undo dispense packs
G137	Bulk undo destroy packs (Not usable as not permitted by the URS)
G147	Bulk undo export packs
G157	Bulk undo sample packs
G167	Bulk undo free sample packs
G177	Bulk undo lock packs
G187	Bulk undo stolen packs (Not usable as not permitted by the URS)
Homogenous Bulk Transactions – Other-Transactions	
G188	Request bulk transaction result
Other	
G199	Request pickup ids for bulk transaction result

Table 17: Homogenous Bulk Transaction use Cases

Important: Despite of the fact that the transactions G137 and G187 are implemented in the system and described below for the sake of completeness, the URS does not foresee the transactions to be used. For that purpose the usage of that transaction is prevented by the rights and roles management of the NMVS.

In the future, if the URS will change and the usage of that particular transaction will become permitted, the transaction will be unlocked.

Currently, please do not implement the transaction in your client software. You will not be able to use it, anyway.

As all relevant bulk services are similar to the single pack services that already has been described the following documentation is an example for 'Bulk Dispense Packs', 'Request Bulk Transaction Result' and 'Bulk Undo Dispense Packs'. For detailed explanation of each service please refer to the single pack chapters at the earlier part of the implantation guideline.

11.3 Homogenous Bulk Transaction – Request Format

Attribute	O / M	Description
B – Body Data		
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs.
Batch expiry data	M	Expiry date, equal for all packs belonging to the same batch.
B4 – Bulk Data		
List of packs	M	List of packs belonging to the bulk operation.
Pack data	M	
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

Table 18: Body – Homogenous Bulk Transaction Request Format

Example:

```
  <Body>
    <Product>
      <ProductCode scheme="GTIN">05060141900015</ProductCode>
      <Batch>
        <Id>1264</Id>
        <ExpDate>171200</ExpDate>
      </Batch>
    </Product>
    <Packs>
      <!--1 to 100000 repetitions:-->
      <Pack sn="XVTR75973491006155" />
      <Pack sn="XVTR75973491006156" />
      <Pack sn="XVTR75973491006157" />
      <Pack sn="XVTR75973491006158" />
    </Packs>
  </Body>
```

11.4 G125 - Bulk Dispense Packs

This use case describes the dispensing of a bulk of packs by a wholesaler or pharmacy who is connected to the national system.

With the ‘Dispense Pack’ use case, the product pack gets decommissioned and the new state becomes “Supplied”.

11.4.1 G125 - Use Case Diagram

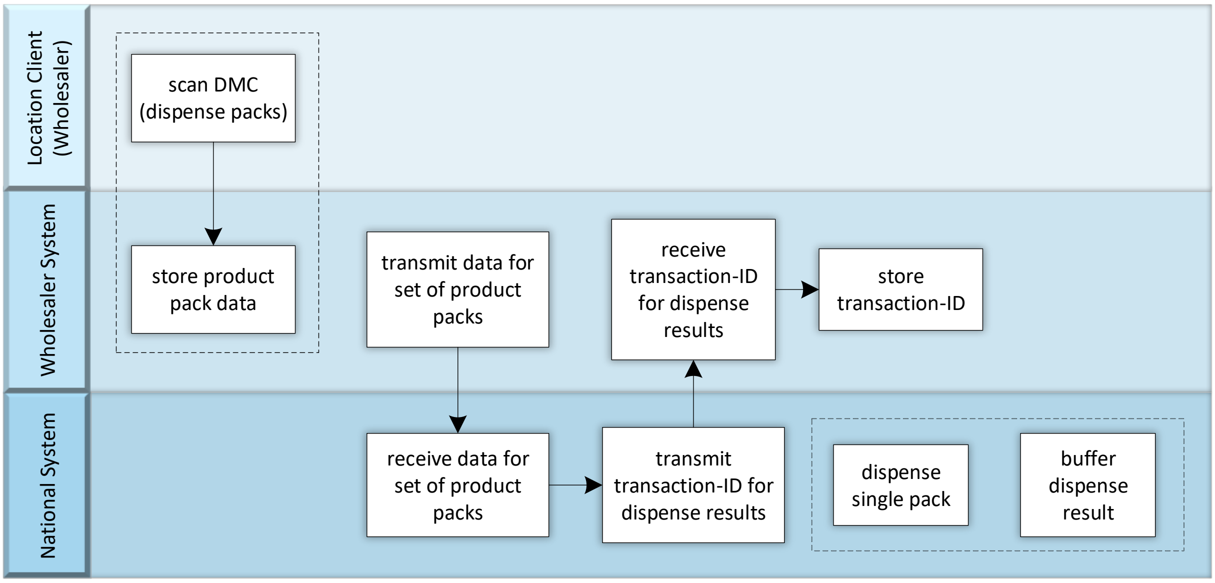

Location Client (Wholesaler)
Wholesaler System
National System
scan DMC (dispense packs)
store product pack data
transmit data for set of product packs
receive transaction-ID for dispense results
store transaction-ID
receive data for set of product packs
transmit transaction-ID for dispense results
dispense single pack
buffer dispense result

Table 19: Bulk Dispense Packs

11.5 G188 - Request Bulk Pack Operation Result

11.5.1 G188 - Process Description

This use case describes how to receive the result of the bulk transaction ‘Dispensing a Bulk of Packs’ by a wholesaler or pharmacy who is connected to the national system.

To get the results the original NMVS transaction ID of the dispensing a bulk of packs needs to be submitted with the request.

11.5.2 G188 - Use Case Diagram

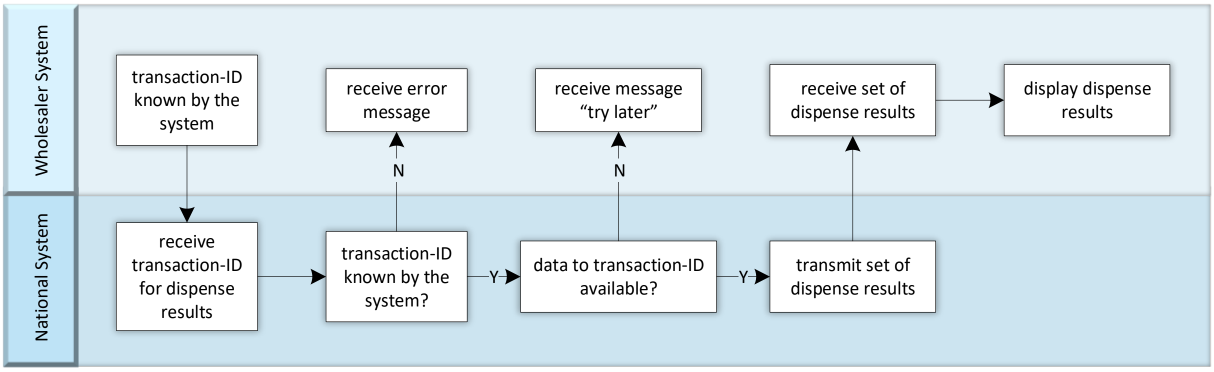

Wholesaler System
National System
transaction-ID known by the system
receive error message
receive message "try later"
receive set of dispense results
display dispense results
receive transaction-ID for dispense results
transaction-ID known by the system?
data to transaction-ID available?
transmit set of dispense results
N
N
Y
Y

Figure 36: Bulk Dispense Pack Operation Results

11.5.3 G188 – Web Service Description

The web service G188 needs to be used to request a bulk pack operation result. For further details, please pay attention to Interface Description for Distributors.xls in the related interface version

11.5.3.1 Bulk Result Transaction – Request Format

Attribute	O / M	Description
B – Body Data		
Reference client transaction ID	M	Unique identifier of the transaction from the client system

Table 20: Body – Bulk Result Transaction Request Format

Example:

```
<Body>
    <RefClientTrxId>a295a850-0daa-4cf1-8e65-c6165c5</RefClientTrxId>
</Body>
```

11.5.3.2 Bulk Result Transaction – Response Format

Attribute	O / M	Description
B – Body Data		
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme.

Attribute	O / M	Description
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
Intermarket flag	O	Informs the caller that the requested pack data was not found in the national database and that therefore an inter-market request was launched.
NHRN	O	National healthcare reimbursement number, optional use in the pharmacy/wholesaler system
Extended master data	O	Reference to extended master data, e.g. leaflet
Product information date	O	Date of the release of this information
Product information changed	O	Indicates if this information has been changed.
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs.
Batch expiry date	M	Expiry date, equal for all packs belonging to the same batch.
NMVS batch expiry date	O	Shows the actual expiry date on NMVS which may be different due to batch update processes
B4 – Bulk Data		
Result list of packs	M	List of packs, either in case of a verification of all packs or all packs that could not be successfully processed
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.
Pack state	M	State of pack

Attribute	O / M	Description
Pack state reasons	O	List of associated detail or reasons why a pack is not able to be dispensed
B6 – Return Code (Pack Level)		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid
B – Body Data		
B6 – Return Code (Body Level)		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 21: Body – Bulk Result Transaction Response Format

Example:

```
<Body>
  <Product>
    <ProductCode scheme="GTIN">05060141900015</ProductCode>
    <!--Optional:-->
    <Intermarket>false</Intermarket>
    <!--Optional:-->
    <NHRN>09556886</NHRN>
    <!--Optional:-->
    <ProductInfoRef date="2015-12-31" modified="false">890754389</ProductInfoRef>
    <!--Optional:-->
    <Batch>
      <Id>1264</Id>
      <!--Optional:-->
      <ExpDate>171231</ExpDate>
      <!--Optional:-->
      <ExpDateNMVS>2017-12-31</ExpDateNMVS>
    </Batch>
  </Product>
  <!--Optional:-->
  <Packs>
```

```
    <!--1 to 100000 repetitions:-->
    <Pack sn="XVTR75973491006155" State="INACTIVE">
      < !--0 to 20 repetitions:-->

      <Reason>SUPPLIED</Reason>
      <Reason>EXPIRED</Reason>
      <Reason>RECALLED</Reason>
      <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
    </Pack>
  </Packs>
  <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
</Body>
```

11.6 G127 - Bulk Undo Dispense Packs

11.6.1 G127 - Process Description

This use case describes the reversing of dispensing a bulk of packs by a wholesaler or pharmacy who is connected to the national system.

With the undo dispense pack use case the product pack gets the state “Active” again.

11.6.2 Undo Bulk Transaction – Request Format

Attribute	O / M	Description
B – Body Data		
Reference client transaction ID	O	Unique identifier of the transaction from client system
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs.
Batch expiry data	M	Expiry date, equal for all packs belonging to the same batch.
B4 – Bulk Data		
List of packs	M	List of packs belonging to the bulk operation.

Attribute	O / M	Description
Pack data	M	
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

Table 22: Body – Undo Bulk Transaction Request Format

Example:

```
<Body>
  <Product>
    <ProductCode scheme="GTIN">05060141900015</ProductCode>
    <Batch>
      <Id>1264</Id>
      <ExpDate>171200</ExpDate>
    </Batch>
  </Product>
  <Packs>
    <!--1 to 100000 repetitions:-->
    <Pack sn="XVTR75973491006155" />
    <Pack sn="XVTR75973491006156" />
    <Pack sn="XVTR75973491006157" />
    <Pack sn="XVTR75973491006158" />
  </Packs>
  <!--Optional:-->
  <RefClientTrxId>c295a450-0caa-4cf1-8e62-c6165c5</RefClientTrxId>
</Body>
```

11.6.3 G127 - Use Case Diagram

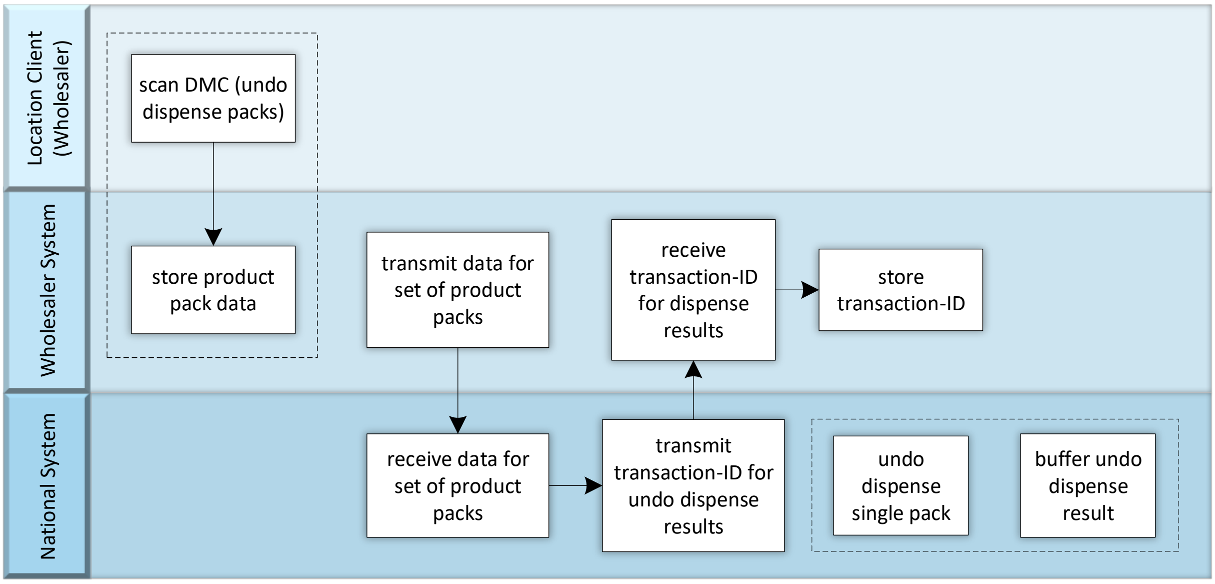

Location Client (Wholesaler)
Wholesaler System
National System
scan DMC (undo dispense packs)
store product pack data
transmit data for set of product packs
receive transaction-ID for dispense results
store transaction-ID
receive data for set of product packs
transmit transaction-ID for undo dispense results
undo dispense single pack
buffer undo dispense result

Figure 37: Bulk Undo Dispense Packs

12 Asynchronous Web Service Operations – Mixed Bulk Transaction

12.1 Process Description

In the unusual event that the NMVS is not reachable due to network problems, the distributor's IT system needs to store the service request for each scanned pack locally in an internal buffer in the given order.

After reestablishing the connection to the NMVS, the distributor's IT system needs to synchronize with the national system. To update the NMVS with the stored scanning events the 'Mixed Bulk' service needs to be used. This service is an asynchronous service. The stakeholder system will be informed about the successful submission with a corresponding NMVS transaction ID. With this transaction ID, the stakeholder system is able to request the results of the 'Mixed Bulk' service from the national system. It is recommended for the stakeholder system to wait until the final results are submitted back before continuing with new scans to avoid any mismatches between the buffered and new scans.

Likely, the connection failure could be based on a broader network malfunction in the region which will be leading to a higher number of later synchronization tasks from the affected region afterwards. It is recommended to send the result request not more often than once in a minute. Any higher frequency will not speed up the process but put a higher load on the system.

All scanning events will be included in the result response afterwards.

Preconditions:

The asynchronous use cases cannot be processed in parallel because of the strict order of the incoming requests. This strict observance of order is important to avoid processing errors and malfunction of the system.

Like the single pack transaction the users need to have a valid certificate to call the Bulk transactions.

12.1 Mixed Bulk Transaction Use Cases

Mixed Bulk transactions are possible for all activities that are allowed within the stakeholder role.

Use Case Code	Business Process
Mixed Bulk Transactions	
G195	Submit mixed bulk transaction
G196	Request mixed bulk transaction result

12.2 G195 – Submit Mixed Bulk Transaction

12.2.1 G195 - Process Description

During the off-line phase the stakeholder system is storing all scanning events in an internal buffer. It is very important for the synchronization afterwards to store the events in the given sequence. After re-storing the connection, the distributor's system transmits the set of product pack data including the type of service to the national system. The national system will then perform the respective service for each of the submitted packs in the submitted sequence. After execution of this service, the national system will temporary internally store the results and send them back on request in one transmission. Each result will be linked with the respective serial number.

This service is an asynchronous service. Depending on the number of products it can take some time before the results will be ready to submit. There are the same checks like in single pack service.

12.2.2 G195 - Submit Mixed Bulk Transaction – Request Format

Attribute	O / M	Description
B – Body Data		
B4 – Bulk Data		
Transaction list	M	List of detail transactions. Transactions must be sorted in ascending order per pack, i.e. oldest transactions first. See below for the structure of the elements of the transaction list.
Transaction item	M	Container for single transaction
Single request item	M/O	Transaction item contains either single request item or single request item manual
Request type not manual	M	Type of request for not manual processes
A3 – Transaction Data		
Client transaction ID	M	Identifies Client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.
Language code	M	Language code, ISO-639-3 compliant.Standard fallback language is English (eng). Supported languages include eng, deu. Further languages may be supported after consultation with the respective NMVO.
Reference client transaction ID	O	Unique identifier of the transaction from the client system.
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs

Attribute	O / M	Description
Batch expiry data	M	Expiry date, equal for all packs belonging to the same batch.
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.
Single request item manual	M/O	Transaction item contains either single request item or single request item manual
Request type manual	M	Type of request for manual processes
A3 – Transaction Data		
Client transaction ID	M	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.
Language code	M	Language code, ISO-639-3 compliant.Standard fallback language is English (eng). Supported languages include eng, deu. Further languages may be supported after consultation with the respective NMVO.
Reference client transaction ID	O	Unique identifier of the transaction from the client system.
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

Table 23: Body – Mixed Bulk Transaction Request Format

Example:

```
<Body>
   <TrxList>
      <!--1 to 100000 repetitions:-->
      <TrxItem>
         <Item reqType="G120">
            <Product>
               <ProductCode scheme="GTIN">05060141900015</ProductCode>
               <Batch>
                  <Id>1264</Id>
                  <ExpDate>171200</ExpDate>
               </Batch>
            </Product>
            <Pack sn="XVTR75973491006155"/>
            <Transaction>
               <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
               <Language>deu</Language>
               <!--Optional:-->
               <RefClientTrxId>a693a450-0caa-4cf1-8e62-c7545c4</RefClientTrxId>
            </Transaction>
         </Item>
      </TrxItem>
      <TrxItem >
         <Item reqType="G121">
            <Product>
               <ProductCode scheme="GTIN">05060141900015</ProductCode>
               <Batch>
                  <Id>1264</Id>
                  <ExpDate>171200</ExpDate>
               </Batch>
            </Product>
            <Pack sn="XVTR75973491006155"/>
            <Transaction>
               <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
               <Language>deu</Language>
            </Transaction>
         </Item>
      </TrxItem>
      <TrxItem >
         <ItemMan reqType="G122">
            <Product>
               <ProductCode scheme="GTIN">05060141900015</ProductCode>
            </Product>
            <Pack sn="XVTR75973491006155"/>
            <Transaction>
               <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
               <Language>deu</Language>
            </Transaction>
         </ItemMan>
      </TrxItem>
   </TrxList>
</Body>
```

12.2.3 G195 - Use Case Diagram

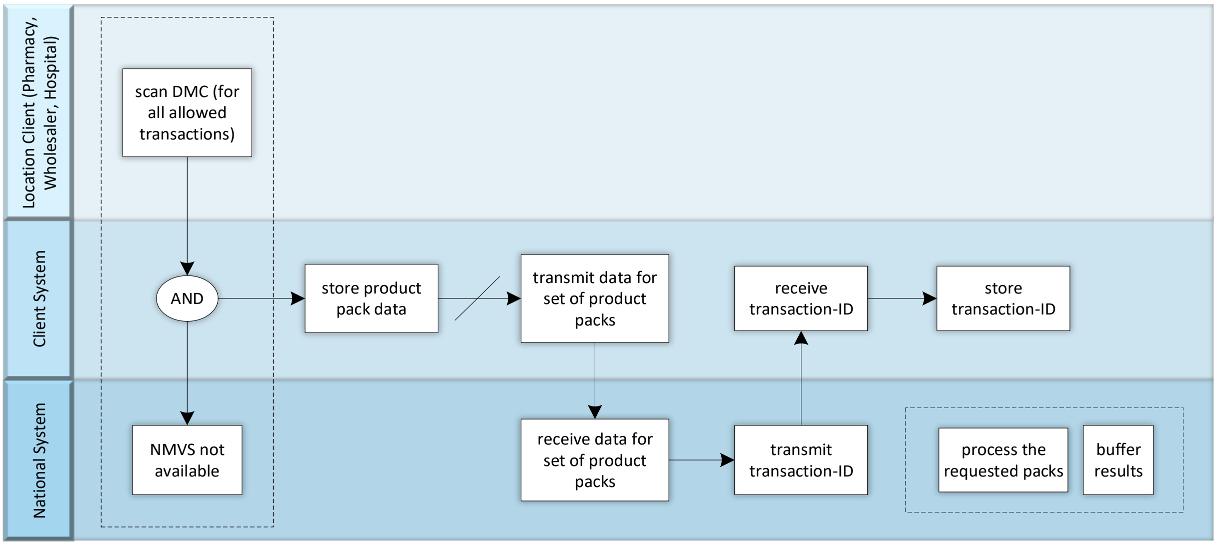

Location Client (Pharmacy, Wholesaler, Hospital)
Client System
National System
scan DMC (for all allowed transactions)
AND
store product pack data
transmit data for set of product packs
receive transaction-ID
store transaction-ID
NMVS not available
receive data for set of product packs
transmit transaction-ID
process the requested packs
buffer results

Figure 38: Offline Concept - Submit Bulk Transaction

12.3 G196 - Request Mixed Bulk Transaction Result

12.3.1 G196 - Process Description

A separate request needs to be submitted to get the results of a Mixed Bulk Transaction.

12.3.2 G196 – Request Mixed Bulk Transaction Result – Request Format

Attribute	O / M	Description
B – Body Data		
Reference client transaction ID	M	Unique identifier of the transaction from the client system

Table 24: Submit Mixed Bulk Transaction – Request Format

Example:

```
<Body>
  <RefClientTrxId>a295a850-0daa-4cf1-8e65-c6165c4</RefClientTrxId>
</Body>
```

12.3.3 G196 - Request Mixed Bulk Transaction Result – Response Format

Attribute	O / M	Description
B – Body Data		
B4 – Bulk Data		
Transaction feedback list	M	List of feedbacks on every transaction of the original transaction list. See below for a description of the structure of a single transaction feedback. For Bulk verify, this list will contain all serial numbers from the request. As for other bulk transactions the list contains only serial numbers which were not processed successfully.
Transaction item	M	Container for single transaction
A3 – Transaction Data		
Language code	M	Language code, ISO-639-3 compliant. Standard fallback language is English (eng). Supported languages include eng, deu. Further languages may be supported after consultation with the respective NMVO.
Client transaction ID	M	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.
NMVS transaction ID	M	Unique identifies of the transaction from the NMVS.
NMVS timestamp	M	Processing timestamp of the NMVSin UTC, marks the end of processing within the NMVS.
B1 – Product Data		
Product code	M	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
Intermarket flag	O	Informs the caller that the requested pack data was not found in the national database and that therefore an intermarket request was launched.
NHRN	O	National healthcare reimbursement number, optional use in the pharmacy/wholesaler system
Extended master data	O	Reference to extended master data, e.g. leaflet

Attribute	O / M	Description
Product information date	O	Product information date
Product information changed	O	Indicates whether this information has been changed.
B2 – Batch Data		
Batch ID	M	ID of the batch to which this pack belongs
Batch expiry data	O	Expiry date, equal for all packs belonging to the same batch.
NMVS batch expiry date	O	Shows the actual expiry date on NMVS which may be different due to batch update processes
B3 – Pack Data		
Pack serial number	M	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.
Pack state	M	State of pack
Pack state reasons	O	List of associated details or reasons why a pack is not able to be dispensed.
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 25: Body – Mixed Bulk Transaction Response Format

Example:

```
  <Body>
```

```
    <!--Optional:-->
    <TrxList>
       <!--0 to 100000 repetitions:-->
      <TrxItem type="G122">
        <Transaction>
          <Language>deu</Language>
          <ClientTrxId>c295a450-0caa-4cf1-8e62-c6165c4</ClientTrxId>
          <NMVSTrxId>c295a450-0caa-4cf1-7e62-c6865d41</NMVSTrxId>
          <Timestamp>2017-12-31T12:00:00</Timestamp>
        </Transaction>
        <Product>
          <ProductCode scheme="GTIN">05060141900015</ProductCode>
          <!--Optional:-->
          <Intermarket>true</Intermarket>
          <!--Optional:-->
          <NHRN>09556886</NHRN>
          <!--Optional:-->
          <ProductInfoRef date="2014-12-31" modified="false">890754389</ProductInfoRef>
          <!--Optional:-->
          <Batch>
            <Id>1264</Id>
            <!--Optional:-->
            <ExpDate>171231</ExpDate>
            <!--Optional:-->
            <ExpDateNMVS>2017-12-31</ExpDateNMVS>
          </Batch>
        </Product>
        <Pack sn="XVTR75973491006155" state="INACTIVE">
          <Reason>SUPPLIED</Reason>
        </Pack>
        <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
      </TrxItem>
    </TrxList>
    <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
  </Body>
</G196Request>
```

12.3.4 G196 - Use Case Diagram

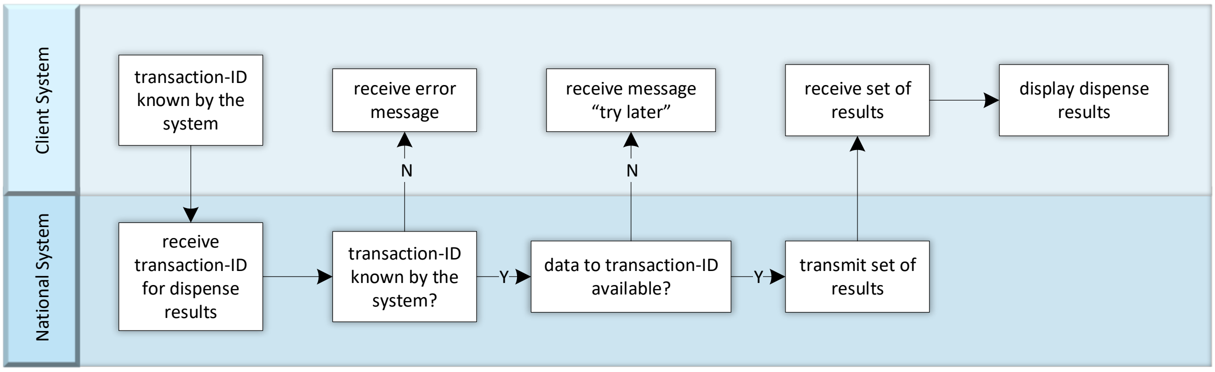

Client System
National System
transaction-ID known by the system
receive error message
receive message "try later"
receive set of results
display dispense results
N
N
receive transaction-ID for dispense results
transaction-ID known by the system?
Y
data to transaction-ID available?
Y
transmit set of results

Figure 39: Request Mixed Bulk Transaction Result

12.3.5 G196 - Web Service Description

The web service G196 for mixed bulk transaction results needs to be used to get the results after reestablishing the connection to the national system and submitting the scanning events via web service G195. The original NMVS transaction ID of the G195 service needs to be submitted with the request to identify the correct results.

13 Other processes

13.1 G101 - Download Product Master Data

13.1.1 G101 - Process Description

The distributor's system is able to download specific information to provide the distributors with the full set of data about the product. This information is the link between the product code that is in the Data Matrix Code and any local registration details like the local reimbursement number. The national system is able to provide this information on request.

13.1.2 G101 - Download Product Master Data – Request Format

Attribute	O / M	Description
B – Body Data		
B4 – Bulk Data		
List of products	M	List of products belonging to the operation.
B1 – Product Data		
Product code	O	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)

Table 26: Body – Download Product Master Data Request Format

Example:

```
<Body>
  <ProductCodeList>
    <!--0 to 1000 repetitions:-->
    <!--If list is empty: return ALL active products: -->
    <ProductCode scheme="GTIN">05060141900015</ProductCode>
  </ProductCodeList>
</Body>
```

13.1.3 G101 - Download Product Master Data - Response Format

Attribute	O / M	Description
B – Body Data		
B4 – Bulk Data		
List of products	M	List of products belonging to the operation.

Attribute	O / M	Description
B1 – Product Data		
Product code	O	Product code in GTIN- or PPN scheme.
Product code scheme	M	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
Product name	M	Name of the product
NHRN	M	National healthcare reimbursement number
MAH ID	M	MAH registration number
List of wholesalers	M	List with names of contracted wholesalers for this product
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	M	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 27: Body – Download Product Master Data Response Format

Example:

```
<Body>
  <ProductList>
      <!--1 to 100000 repetitions:-->
      <Product>
         <ProductCode scheme="GTIN">05060141900015</ProductCode>
         <Name>Tripapolon</Name>
         <NHRN>12345</NHRN>
         <MahId>6789</MahId>
         <Wholesalers>
            <!--0 or more repetitions:-->
            <Wholesaler>Example</Wholesaler>
         </Wholesalers>
      </Product>
   </ProductList>
   <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed"/>
</Body>
```

13.1.4 G101 - Use Case Diagram

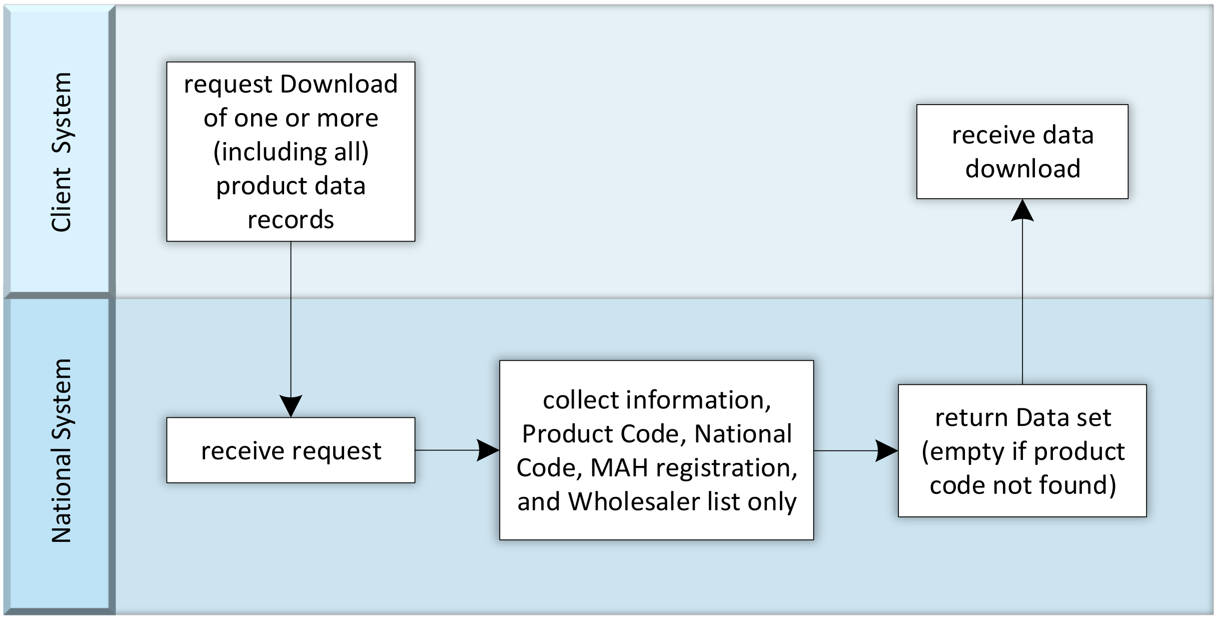

Client System
National System
request Download of one or more (including all) product data records
receive data download
receive request
collect information, Product Code, National Code, MAH registration, and Wholesaler list only
return Data set (empty if product code not found)

Table 28: Download Product Master Data

13.1.5 G101 - Detailed Process Description

A request for product master data download is coming from the distributor's system. It can be for a single product code, a set of product codes or all product codes that are stored as records within the national system. The national system receives the request. The returned product code records are:

1. The product code
2. The national product code (reimbursement number)
3. The MAH registration number
4. The list of contracted wholesalers (for this product code)

The set of data is returned to the requesting client system and if the requested product code in question is not found, the data set for that code will be returned as empty. The requesting distributor's system will receive the pieces of information and has to store these internally.

13.2 G199 - Request Pickup IDs for Bulk Transaction Result

13.2.1 G199 - Process Description

This use case describes how to receive the results of bulk transactions of various pack services by a wholesaler or pharmacy who will be connected to the national system if the original NMVS transaction ID is lost.

In that case, the distributor IT system can request a list of all original NMVS transaction IDs of requests that were not processed yet.

Normally, the distributor's IT system needs to provide the transaction ID of the initial transaction to get the result response. If it is not possible to provide this transaction ID, the user will be able to process the 'Pickup IDs for Bulk Result' transaction to get a list of transaction IDs. The list will give users the possibility to identify which transactions are still open to collect the result.

This will mostly be useful if a distributor's IT system is in an inconsistent state.

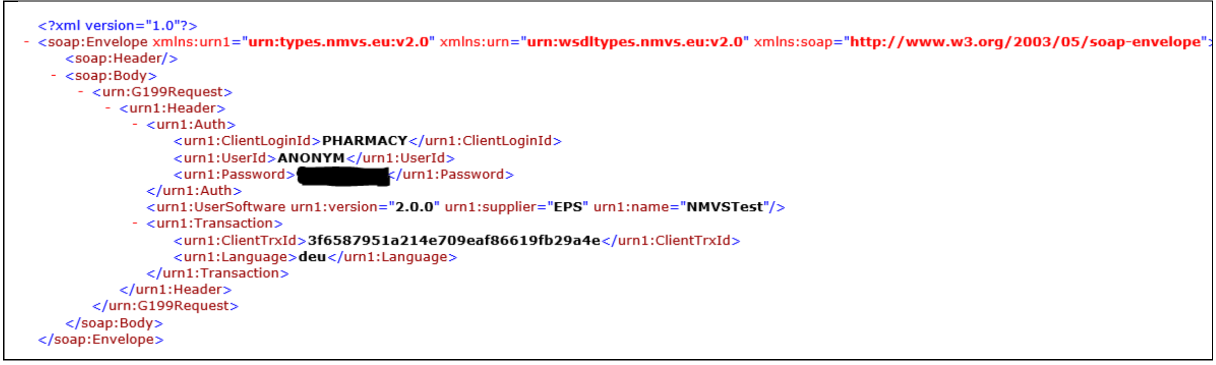
13.2.2 G199 - Pickup IDs for Bulk Result – Request Format

The request format of the pickup IDs transaction only consists for the header elements which are described in detail in 8.4.1.1.

Attribute	O / M	Description
A3 – Transaction Data		
Client transaction ID	M	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.

Table 29: Pickup IDs for Bulk Result – Request Format

Example:



```
  <?xml version="1.0"?>
- <soap:Envelope xmlns:urn1="urn:types.nmvs.eu:v2.0" xmlns:urn="urn:wsdltypes.nmvs.eu:v2.0" xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
    <soap:Header/>
  - <soap:Body>
    - <urn:G199Request>
      - <urn1:Header>
        - <urn1:Auth>
            <urn1:ClientLoginId>PHARMACY</urn1:ClientLoginId>
            <urn1:UserId>ANONYM</urn1:UserId>
            <urn1:Password>[redacted]</urn1:Password>
          </urn1:Auth>
          <urn1:UserSoftware urn1:version="2.0.0" urn1:supplier="EPS" urn1:name="NMVSTest"/>
        - <urn1:Transaction>
            <urn1:ClientTrxId>3f6587951a214e709eaf86619fb29a4e</urn1:ClientTrxId>
            <urn1:Language>deu</urn1:Language>
          </urn1:Transaction>
        </urn1:Header>
      </urn:G199Request>
    </soap:Body>
  </soap:Envelope>
```

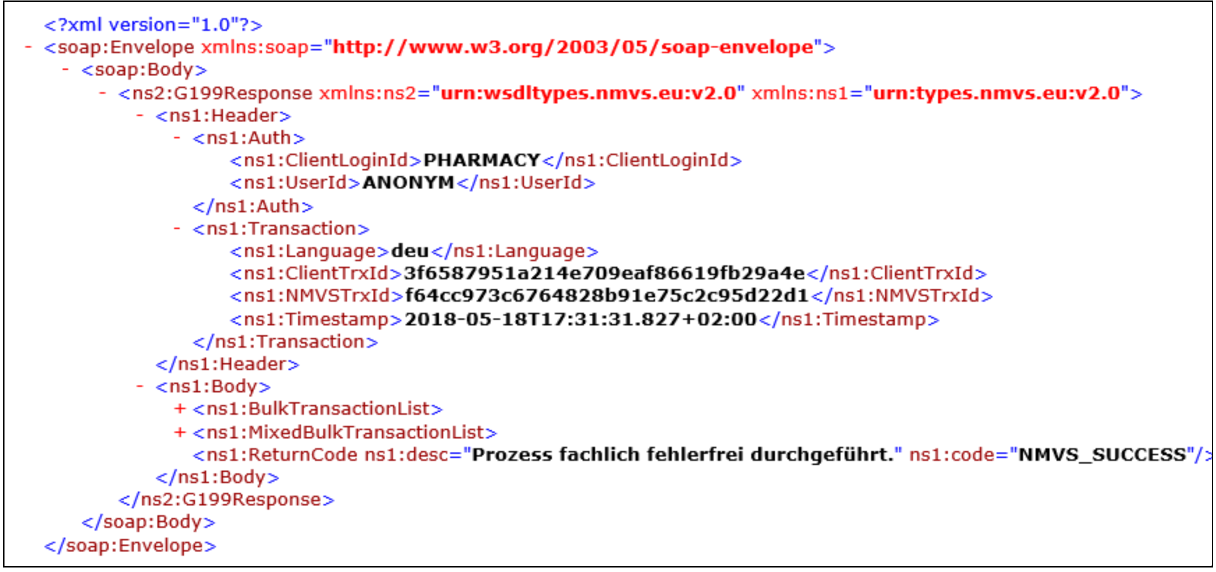
13.2.3 G199 - Pickup IDs for Bulk Result – Response Format

Attribute	O / M	Description
B – Body Data		
A3 – Transaction Data		
Client transaction ID	M	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.
NMVS transaction ID	M	Unique identifier of the transaction from the NMVS System.
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21

Attribute	O / M	Description
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 30: Body – Bulk Result Transaction Response Format

Example:



```
  <?xml version="1.0"?>
- <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  - <soap:Body>
    - <ns2:G199Response xmlns:ns2="urn:wsdltypes.nmvs.eu:v2.0" xmlns:ns1="urn:types.nmvs.eu:v2.0">
      - <ns1:Header>
        - <ns1:Auth>
            <ns1:ClientLoginId>PHARMACY</ns1:ClientLoginId>
            <ns1:UserId>ANONYM</ns1:UserId>
          </ns1:Auth>
        - <ns1:Transaction>
            <ns1:Language>deu</ns1:Language>
            <ns1:ClientTrxId>3f6587951a214e709eaf86619fb29a4e</ns1:ClientTrxId>
            <ns1:NMVSTrxId>f64cc973c6764828b91e75c2c95d22d1</ns1:NMVSTrxId>
            <ns1:Timestamp>2018-05-18T17:31:31.827+02:00</ns1:Timestamp>
          </ns1:Transaction>
        </ns1:Header>
      - <ns1:Body>
        + <ns1:BulkTransactionList>
        + <ns1:MixedBulkTransactionList>
          <ns1:ReturnCode ns1:desc="Prozess fachlich fehlerfrei durchgeführt." ns1:code="NMVS_SUCCESS"/>
        </ns1:Body>
      </ns2:G199Response>
    </soap:Body>
  </soap:Envelope>
```

13.2.4 G199 - Use Case Diagram

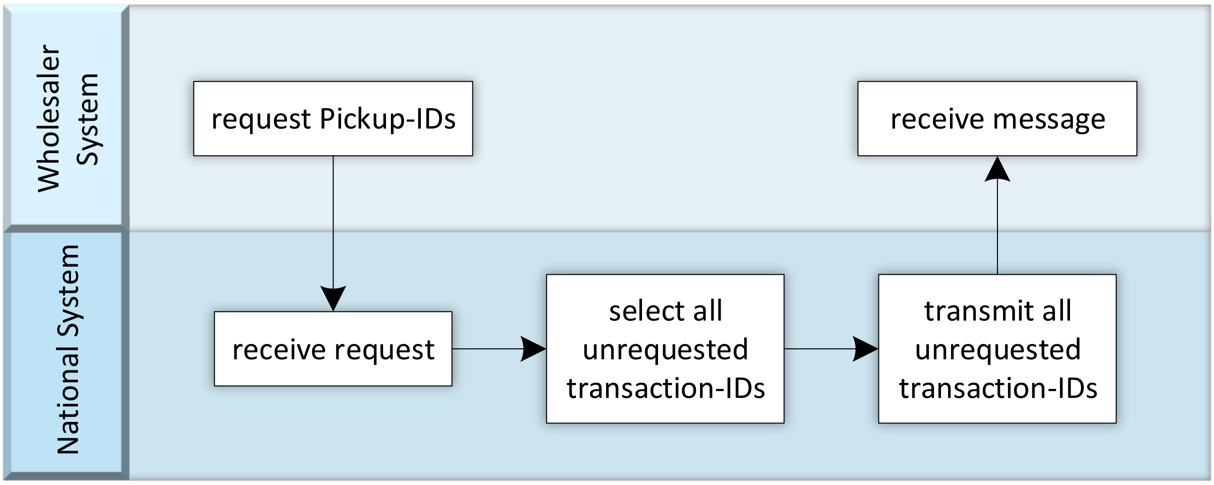


Figure 40: Request Pickup IDs for Bulk Transaction Result

13.3 Change Password from Pharmacy / Wholesaler

This use case describes the change of an existing password. It is used by pharmacists and wholesalers.

The distributor’s system needs to inform the operator in advance that the password will expire soon. After expiration, the system will force the operator to change the password immediately.

13.3.1 G445 - Process Description

Due to security regulation the user needs to change the password that is used to connect to the national system via web services as well as the fallback scenario via Web GUI access from time to time. The operator will type in the new password. The new information will be sent to the national system for processing. The national system will check that the submitted user credentials (client ID, user ID and current password) are correct and if the new password meets the given requirements regarding format/range (mix of alpha and numeric character, upper / lower case and at least one non-alpha / non-numeric character) and length (min of 8 characters and digits). The second check will be if the new password was used within the last 20 used passwords. If all checks are correct the national system will update the password, an event of the password change action is recorded within the national system and a message confirming the successful change is sent back to the local IT system to display the success to the operator.

If the checks are not correct, the change password process will be rejected and a message will be sent back to the IT system.

13.3.2 G445 - Password Change Process

Ten days before the final expiry of the current password, the NMVS starts to send out notifications to the client software with each response to web service requests. The used notification is NMVS_NOTIFY_AU_02, the description is :

> “Your password will expire in *xyz* day(s).
> Please change your password using the password-change function in the system.
> If your password expires, you will no longer be able to connect to the system.”

Where *xyz* will be replaced with the actual number of days left. The NMVO that is running the NMVS can adopt the description to the local language, so it is recommended for the client software to present the description directly to the user.

Once the password expired, no transactions except for G445 are usable for the affected user, for that reason, G445 should be called before the expiry. There is no limit when a new password can be set at the earliest, you don’t need to wait for the expiry notification from the system.

Using G445 the password can be only set for the user calling. It is not possible to use G445 or any other transaction to set a new PW for other users. When calling G445 the current (old) password is to be used in the header, both passwords (old + new) are to be provided in the request body.

Once the G445 transaction is performed, the new password is immediately valid, still the old password remains valid for five further days, too. When using the old password during these five days, the server will notify the client software with NMVS_NOTIFY_AU_01 with each response:

> “You are using an old password.
> The password of your user account has changed, so please adjust your password as soon as

possible.
The password you are using will be deactivated in *xyz* day(s).”

Where *xyz* is the number of remaining days which the old password remains valid.

Leaving the old password valid for some additional days comes in handy when multiple client systems are using the same user name and password, for example multiple checkout counters within one pharmacy. Once one of the checkout counters has set a new password on the server and for its own use, the other counters can catch up changing the password they are using, but they won’t be disconnected immediately. Still all the other systems will receive the notification NMVS_NOTIFY_AU_01 until they start to use the new password, too.

While using the old password during these five days, all transactions are usable except for G445, which means in the case of the multiple checkout counters, they can continue working with the old password for a while, but they can’t re-change the newly set password.

13.3.3 G445 - Web Service Description

The web service G445 needs to be used to change a password. For further details, please pay attention to Interface Description for Distributors.xls in the related interface version

13.3.4 G445 - Change Password – Request Format

Attribute	O / M	Description
B – Body Data		
Password	M	Password to authenticate a user.
New Password	M	New password to authenticate a user. This password has to be different from the previous password.

Table 31: Body – Change Password Request Format

Example:

```
<Body>
  <Password>838hdjLk#</Password>
  <NewPassword>272sahSd#</NewPassword>
</Body>
```

13.3.5 G445 - Change Password – Response Format

Attribute	O / M	Description
B – Body Data		
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21

Attribute	O / M	Description
Return code description	M	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 32: Body – Change Password Response Format

Example:

```
<Body>
  <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
</Body>
```

13.4 G615 - Download Certificate for Pharmacy / Wholesaler

This use case describes the download process of an initial or new certificate (for example to replace an old expiring certificate). It is used by pharmacists and wholesalers, but also for all other stakeholders who need a certificate to communicate or get access to the NMVS.

In case a certificate expires, the distributor's system will no longer be able to connect to the national system.

13.4.1 G615 - Process Description

Due to security regulations the user needs to renew the certificate to connect to the national system via web services as well as the fallback scenario via Web GUI access from time to time. The certificate download request should be sent to the national system for processing. The national system will submit the new certificate to the requestor and an event of the download action is recorded within the national system.

Before a new certificate can be download, it needs to be created. There are two possible ways for creation of a new certificate in the NMVS (the creation of a certificate can only be triggered by administrators):

1. The client or NMVO administrator creates a new certificate manually for the user via Web GUI (e.g. the user has lost his certificate)
   a. The NMVO administrator is able to create a new certificate for ALL users in the NMVS
   b. The client administrator is able to create a new certificate only for the users in his client
2. The certificate is about to expire soon, so that a renewal of the certificate is required (renewal of a certificate always means the replacement of the old with a new certificate). For further information, please refer to the next section.

13.4.2 G615 - Certificate renewal process

The national system will automatically generate a new certificate 30 days before the expiration of the certificate. On the same day, it will send a notification e-mail to all users, whose certificates are about to expire. In this e-mail, the users will receive all the necessary information about the new certificate they need to download it via the PKI-Portal or web service. Afterwards, the certificate can be implemented into the distributor's system environment. If the user doesn't download his new certificate within the next 30 days, he will get a daily reminder mail, until the certificate is downloaded.

Based on the two-factor authentication the system requires a valid certificate besides user name and password for all creation requests of new certificates. The actual download request will only require the user/password authentication, it is not necessary to deliver the certificate additionally.

13.4.3 G615 - Download via Web Service (for Distributor Systems)

The distributor's system has to submit the download request via web service to the national system. The national system will check the request data and will reply with the new certificate information (refer to Table 34).

To process the download transaction, the user needs a TAN which will be delivered in the certificate email. If, for any reasons, the password is lost or the user account gets locked (NMVS _FE_AU_02), it will be possible to initiate a recovery by contacting the responsible client administrator or the NMVO.

An additional precaution could be, that the distributor's system check the expiration date of the installed client certificate frequently itself. Before expiration, the system should inform the distributor that a renewal of the certificate is needed. The distributor should confirm the process to download and install the new certificate.

⚠ The description for downloading a certificate via PKI-Portal is not part of the implementation guideline. For more information, please contact the responsible NMVO.

13.4.4 G615 - Download Certificate – Request Format

Attribute	O / M	Description
B – Body Data		
B5 - Administration Data		
TAN	M	A TAN number to identify a client certificate

Table 33: Body – Download Certificate Request Format

Example:

```
<Body>
  <Tan>126334</Tan>
</Body>
```

13.4.4.1 Download Certificate – Response Format

Attribute	O / M	Description
B – Body Data		
B5 - Administration Data		
Certificate	M	Binary data of the private certificate to identify the user.
Certificate name	M	Name of the .p12 file containing the private certificate of user

Attribute	O / M	Description
Certificate password	M	Password to access and install the private certificate.
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 34: Body – Download Certificate Response Format

Example:

```
<Body>
  <Cert name="John002" passphrase="HJ7HK8BKAD0QLIN">
    VBERi0xLjMKJaqrrK0KNCAwIG9iago8PCAvVHlwZSAvSW5mbw...
  </Cert>
  <ReturnCode code="NMVS_SUCCESS" desc="Successfully processed" />
</Body>
```

13.5 Terms and Conditions

As a country specific option, the NMVS offers a functionality called “Terms and Conditions”. As a software supplier, please check with the local NMVO whether Terms and Conditions will be managed by the NMVS.

Terms and Conditions, or *T&Cs* for short, represent a contract between an end user (e.g. a pharmacy or a wholesaler) and the NMVO. In this context, the NMVS can be used as a medium between the NMVO and the end users to present T&Cs to end users and having them formally accepted by the end users. The role of the end user software in such a scenario is to support this process.

T&Cs are subject to versioning, i.e. there is a contiguous sequence of T&Cs such that there is never more than one current T&Cs version at any moment in time.
Note: Initially, it is possible that no T&Cs may be defined at all. This status may persist as long as the NMVO decides.

Whenever a new T&Cs version is introduced, it is assigned a date when its validity period starts (this date is usually in the future, but may also be set to a date in the past, meaning that the T&Cs become effective immediately). Once the newest T&Cs version is activated, the preceding T&Cs version (if one exists) expires.

If T&Cs have been issued by the NMVO via the NMVS, each end user is required to accept at least one version of the T&Cs. With each new T&Cs version, each user is granted a *grace period* during which the NMVS can be used without accepting the new T&Cs (again, the grace period may be negative, meaning that the T&Cs have to be accepted immediately). The end of the grace period may vary depending on the user.

T&Cs may be *mandatory* or *optional*. If a T&C version is mandatory, every user needs to accept it before the end of the grace period.

If a T&Cs version is optional, the end user may decide not to accept it – as long as he has already accepted a previous version. If the user has not yet accepted a previous version and the T&C version is set as 'optional', then it is still treated as 'mandatory' for this user. In the case where the end user has accepted a previous T&Cs version, the user may access the NMVS even beyond the grace period .

For users joining the NMVS at a later time (e.g. 6 months after the initial onboarding), the current T&Cs version is mandatory in any case.

NOTE: Failure to accept mandatory T&Cs before the grace period ends will block the user from submitting any transaction requests until acceptance, i.e. such requests will be rejected with error code NMVS_FE_AU_23 (see Table 32 below).

If the T&Cs functionality is active in an NMVS, every end user software connecting with that NMVS has to support the following requirements 1 & 2:

1. As long as there are T&Cs available for acceptance by a user, the NMVS sends a notification with each transaction (see Table 36). Each notification consist of a notification code and a notification text, see Table 35.
   See the following XML fragment for how a notification looks like:

```
<Notification>
   <!-- List of notifications: -->
   <NotificationCode>NMVS_NOTIFY_TC_01</NotificationCode>
   <NotificationDesc>A new version of the NMVS Terms and Conditions is available.
Please accept the new terms and conditions within the next 5 day(s).
If you do not accept the new terms and conditions version in time, you will be denied
system access until acceptance.
  </NotificationDesc>
</Notification>
```

   This leads to *requirement 1:*
   The end user software has to display the notification code and text at least once a day in such a way that the end user takes notice.
   Note: Notification codes NMVS_NOTIFY_PMS_02 and NMVS_NOTIFY_PMS_03 are sent when the acceptance of the latest T&Cs is optional, so these notifications may be displayed with a lesser urgency and/or reduced frequency .

2. The end user must be given the option to read and accept the current T&Cs. For this purpose, two web services are available:

13.5.1 G482 Load Terms and Conditions

Use this web service to download the current Terms and Conditions and to display them within your end user software.

Note: Terms and Conditions are provided as a single plain text string with only basic formatting (e.g. line breaks, titles in capitals). The text may be quite long, though, therefore a display window with a scroll bar is required. It shall also be possible to copy the text (about T&C) from the display window, e.g. to paste it into a text editor and print it out. Together with the text, the version number of the T&Cs and their acceptance status (as returned by G482) should also be displayed.

```
<body>
   <TermsAndConditions>
```

```
      <Version>2</Version>
      <Text>A very long plain text containing the Terms and Conditions with clauses for
each and every imaginable situation. May be repeated in several languages.
1. Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt
ut labore et dolore magna aliqua.
2. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid
ex ea commodi consequat.
3. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla
pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia
deserunt mollit anim id est laborum.
      </Text>
      <!-- Start date means 00.00h local NMVS time of the specified date YYMMDD: -->
      <StartDate>190331</StartDate>
      <Accepted>false</Accepted>
   </TermsAndConditions>
</Body>
```

13.5.2 G483 Confirm Terms and Conditions

Use this web service to inform the NMVS that the current T&Cs have been accepted.

Hence *requirement 2* is:
The end user software has to implement calls for G482 and G483 in order to display and confirm Terms and Conditions under the following circumstances:

I. ***A web service request returns error code NMVS_FE_AU_23.***
This requires immediate acceptance of the current T&Cs.
II. ***A notification NMVS_NOTIFY_PMS_01 is returned.***
This requires the end user software to display the notification at least once a day and to offer the possibility to display the current T&Cs and to accept or skip them.
III. ***A notification NMVS_NOTIFY_PMS_02 or 03 is returned.***
This requires the end user software to display the notification and to offer a possibility to display the T&Cs and to accept or skip them.

Note: See file TD-001 (G100 Distributor Transactions) for more detailed examples.

Notification Code	Notification Text
NMVS_NOTIFY_PMS_01	A new mandatory version of the NMVS Terms and Conditions is available. Please accept the new Terms and Conditions within the next $N day(s). If you do not accept the new Terms and Conditions in time, you will lose system access until acceptance.
NMVS_NOTIFY_PMS_03	A new version of the NMVS Terms and Conditions is available. Please accept the new Terms and Conditions within the next $N day(s).

Table 35: List of Notification Codes Referring to T&Cs.

A4 - Notifications	An optional field which provides additional information to the end user under certain circumstances.
Notification code	Alphanumeric notification code, e.g. NMVS_NOTIFY_AU_01.
Notification description	Textual description of the notification code, e.g. reminder of password expiration.

Table 36: Notification Field as Part of NMVS Web Service Output.

See file *TD-001 (G100 Distributor Transactions)* for more details.

Error Code	Message Text
NMVS_FE_AU_23	The Terms and Conditions need to be accepted immediately.

Table 37: List of Error Codes Referring to T&Cs.Web GUI

Additionally, there is the possibility for pharmacies to verify and dispense a pack manually via a Web GUI (graphical user interface), too.

The Web GUI usage will be restricted after the due date of the legislation. Afterwards, the usage of the Web GUI is only permitted in case the pharmacy software has a malfunction. The GUI is not allowed to be used as the mainly preferred system.

To use the Web GUI, the pharmacist needs to manually type in the serial number and the product number of the pack in hand. It is not permitted to connect a scanner to the GUI, nor to copy and paste the serial number and product number from another application into the input fields of the Web GUI.

It is mandatory that the pharmacy is on boarded to use the Web GUI. The pharmacy needs to be known to the national system. A certificate given by the national system needs to be installed. The pharmacy needs to have a valid user and a password that is known to the national system. Without this preconditions the usage of the Web client is not possible. See also chapter 6.1**.**

If the pharmacy is connected via a centralized access point like a pharmacy association hub, it will be possible to hide all pharmacies/wholesalers behind this central server. In that case the pharmacy in question needs to register separately for the usage of the Web GUI to receive account information including user-ID and password. In this environment, there is no link between the access via the pharmacy system and the Web GUI. Undo services for transactions done via the pharmacy system are not executable via the Web GUI due to the missing link between both interfaces.

You get access to the Web client via the secure web link mentioned in the onboarding documentation that was submitted by letter

13.6 Web GUI

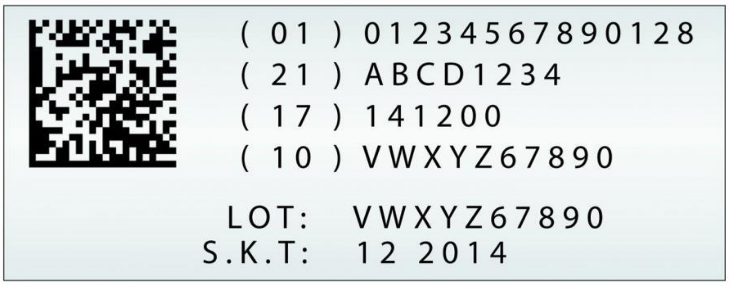
13.6.1 Web GUI Login

The GUI functionality is rudimentary and focused on easy usage and on the main and crucial information.

The operator needs to log in to the NMVS Web GUI with the registered user ID and password. After having logged in successfully, there are only two possible functions: To verify a pack and to dispense a pack. All the other functions that are possible with the distributor's IT system are not implemented in the Web GUI.

13.6.2 Web Client Verify

With the verify functionality the operator is able to check the validity of a pack. Two manual inputs are needed to identify the pack to use this functionality. Therefore, the operator needs to type in the information that are human readable on the pack in hand. Below, there is an example of possible coding on the pack.



(01) 01234567890128
(21) ABCD1234
(17) 141200
(10) VWXYZ67890

LOT: VWXYZ67890
S.K.T: 12 2014

Figure 41: Pack example with serialization

There could be different ways of coding data in human readable form to identify the needed information. In the given case, the identifiers are coded and not directly written. The product number is coded with the identifier '(01)' and the input for the web GUI is '01234567890128'. The serial number is meant by the identifier '(21)' and the input for the web GUI is 'ABCD1234'. The identifiers are standardized in that case based on GS1 standards. Depending on the pharmaceutical producer there is as well the opportunity to mark the respective information with plaintext like 'GTIN:' for the product number and 'SERIAL:' or 'SN:' for the serial number like in the given example with 'LOT:' for the batch ID.

After the submission of the product and serial number to the national system, the execution within the NMVS is similar to the web service to verify a single pack (see chapter 10.1).

If the input of product and serial number is correct and the pack is known to the national system, the results will be presented to the operator. The following information will be presented:

Attributes	Description
Product code	Product code in GTIN- or PPN scheme. The details to the two types are in "Data Type" in the next two rows.
Product code scheme	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
Pack serial number	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.

Attributes	Description
Batch ID	ID of the batch to which this pack belongs.
Batch expiry date	Expiry date, equal for all packs belonging to the same batch.
NHRN	National healthcare reimbursement number, optional use in the pharmacy/wholesaler system
Pack state	State of a pack, which could be either ‘Active’ when the pack is able to be dispensed or ‘Inactive’ in case the pack is not dispensable. If the state is ‘Inactive’, at least one pack state reason has to be provided.
Intermarket flag	Informs the caller that the requested pack data were not found in the national database and that therefore an intermarket request was launched.
NMVS batch expiry date	Shows the actual expiry date on NMVS which may be different due to batch update processes.

There is no status change of the pack in the NMVS.

If the pack is not known, an error message will be presented to the operator. The operator should check the input to make corrections and submit again.

13.6.3 Web Client Dispense

With the dispense functionality the operator is able to dispense a pack. Two manual inputs are needed to identify the pack to use this functionality. Therefore, the operator needs to type in the information that is human readable on the pack in hand. Figure 38 shows an example of possible coding on the pack.

There could be different ways of coding in human readable form to identify the needed information. In the given case, the identifiers are coded and not directly written. The product number is coded with the identifier ‘(01)’ and the input for the web GUI is ‘01234567890128’. The serial number is meant by the identifier ‘(21)’ and the input for the web GUI is ‘ABCD1234’. The identifiers are standardized in that case based on GS1 standards. Depending on the pharmaceutical producer there is as well the opportunity to mark the respective information with plaintext like ‘GTIN:’ for the product number and ‘SERIAL:’ or ‘SN:’ for the serial number like in the given example with ‘LOT:’ for the batch ID.

After the submission of the product and serial number to the national system the execution within the NMVS is similar to the web service G120 - Web Service Description

The web service G120 is used to dispense a single pack. Please pay attention to document Interface Description for Distributors.xls in the related interface version for further details.

G122 - Dispense Single Pack – Manual Data Entry (see also chapter 10.1.5.3).

If the input of product and serial number is correct and the pack is known to the national system, the results will be presented to the operator. Following information will be presented:

Attributes	Description
Product code	Product code in GTIN- or PPN scheme. The details to the two types are in "Data Type" in the next two rows.
Product code scheme	Name of the coding system for product code (GTINs and PPNs are possible in any country due to inter-market transactions)
Pack serial number	Identifies a single pack together with the product ID. Remark: Different packs may have the same serial number if their product IDs are different. Packs with a common product identifier must have different serial numbers even if they have different batch IDs.
Batch ID	ID of the batch to which this pack belongs.
Batch expiry date	Expiry date, equal for all packs belonging to the same batch.
NHRN	National healthcare reimbursement number, optional use in the pharmacy/wholesaler system
Pack state	State of a pack, which could be either ‘Active’ when the pack is able to be dispensed or ‘Inactive’ in case the pack is not dispensable. If the state is ‘Inactive’, at least one pack state reason has to be provided.
Intermarket flag	Informs the caller that the requested pack data were not found in the national database and that therefore an intermarket request was launched.
NMVS batch expiry date	Shows the actual expiry date on NMVS which may be different due to batch update processes.

With the dispense pack use case, the product pack gets decommissioned and the new state “Supplied”.

If the pack is not known, an error message will be presented to the operator. The operator should check the input to make corrections and submit again.

14 QA for Distributors Systems

The NMVS concept has not foreseen a dedicated QA procedure for the individual distributors' IT systems. Before a stakeholder system is able to connect to the national system, there is a number of preconditions to be fulfilled. A software supplier for distributors' IT systems has to run through an onboarding procedure as well as the pharmacies and wholesalers. After the successful onboarding, the software supplier gets access to the related documentation to implement the needed functionality regarding fulfillment of the legislation. Software suppliers will get test material and access to the test system as well. There is a defined procedure to get acceptance of the new software version. Only a full tested and accepted software version should be installed in the stakeholder environment. Only such a version is allowed to connect to the national system.

Arvato provides a guideline for "Client Software Baseline Tests" which can be utilized as it is, reduced, extended and adopted by any means by the NMVOs. The baseline tests are based on several test cases to be performed by the client software. Either, by self-validation or validation by the NMVO, the software supplier needs to prove, the software is capable of properly utilizing the interface.

For all end users, there will be no dedicated service and support offered by Arvato Systems for the stakeholder connection to the NMVS. NMVO will be in charge of the setup of the infrastructure and services related to support. Please contact your local NMVO regarding your service and support.

15 Security

It is mandatory for every user to register with a valid email address to ensure communication of one-time passwords, download links, TANs as well as security reminders.

The NMVS API is protected by three layers of security:

1. Mandatory user ID, member ID and password
2. Client-side request signing using the NMVS-issued Certificate
3. Transport Layer Security (TLS) data transport.

If authentication fails at one of these layers, access to the NMVS API service will be denied.

15.1 User and Password

For security reasons the NMVS must verify that users are permitted to initiate a transaction. If the request cannot be authenticated a SOAP security fault will be returned.

The NMVS is secured by user/password credentials. User accounts as well as all user permissions within any given business processes are bound to clients, not to be mistaken with the client systems. Typically, the permissions are granted to the clients so that users can call the API.

In the SOAP request body, any SOAP client must provide valid values for *ClientLoginID*, *UserID* and *Password* in the *Authentication* section.

The password must meet the following criteria:

- its length must be between 8 and 60 characters
- it must contain at least one lower case letter
- it must contain at least one upper case letter
- it must contain at least one digit
- it must contain at least one special character

Regular expression for valid password
((?=.*\d)(?=.*[a-z])(?=.*[A-Z])(?=.*[\W_]).{8,60})

Table 38: Regular Expression for Password Validation

15.2 Change Password

15.3 If the NMVS returns *NMVS_FE_AU_11,* the user password has expired and needs to be changed via G445 Request, see 13.2- G199 - Request Pickup IDs for Bulk Transaction Result

15.3.1 G199 - Process Description

This use case describes how to receive the results of bulk transactions of various pack services by a wholesaler or pharmacy who will be connected to the national system if the original NMVS transaction ID is lost.

In that case, the distributor IT system can request a list of all original NMVS transaction IDs of requests that were not processed yet.

Normally, the distributor’s IT system needs to provide the transaction ID of the initial transaction to get the result response. If it is not possible to provide this transaction ID, the user will be able to process the ‘Pickup IDs for Bulk Result’ transaction to get a list of transaction IDs. The list will give users the possibility to identify which transactions are still open to collect the result.

This will mostly be useful if a distributor’s IT system is in an inconsistent state.

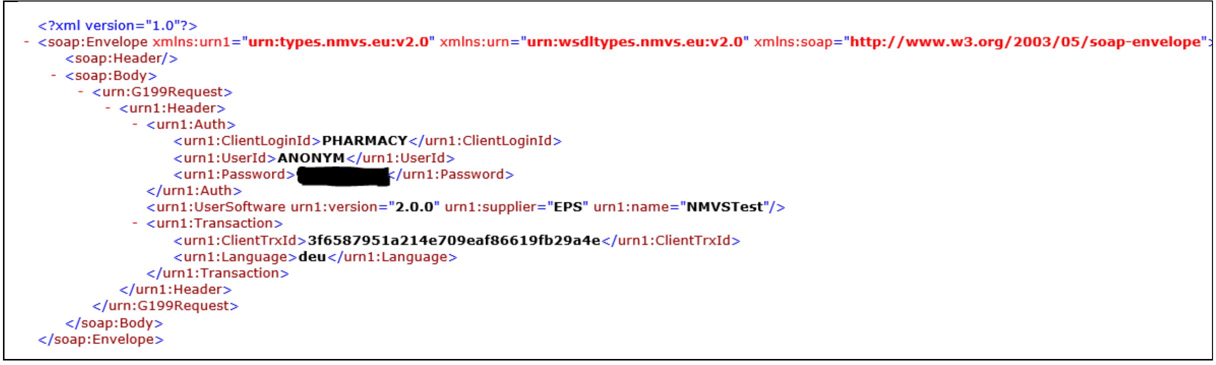
15.3.2 G199 - Pickup IDs for Bulk Result – Request Format

The request format of the pickup IDs transaction only consists for the header elements which are described in detail in 8.4.1.1.

Attribute	O / M	Description
A3 – Transaction Data		
Client transaction ID	M	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.

Table 29: Pickup IDs for Bulk Result – Request Format

Example:



```
  <?xml version="1.0"?>
- <soap:Envelope xmlns:urn1="urn:types.nmvs.eu:v2.0" xmlns:urn="urn:wsdltypes.nmvs.eu:v2.0" xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
    <soap:Header/>
  - <soap:Body>
    - <urn:G199Request>
      - <urn1:Header>
        - <urn1:Auth>
            <urn1:ClientLoginId>PHARMACY</urn1:ClientLoginId>
            <urn1:UserId>ANONYM</urn1:UserId>
            <urn1:Password>[redacted]</urn1:Password>
          </urn1:Auth>
          <urn1:UserSoftware urn1:version="2.0.0" urn1:supplier="EPS" urn1:name="NMVSTest"/>
        - <urn1:Transaction>
            <urn1:ClientTrxId>3f6587951a214e709eaf86619fb29a4e</urn1:ClientTrxId>
            <urn1:Language>deu</urn1:Language>
          </urn1:Transaction>
        </urn1:Header>
      </urn:G199Request>
    </soap:Body>
  </soap:Envelope>
```

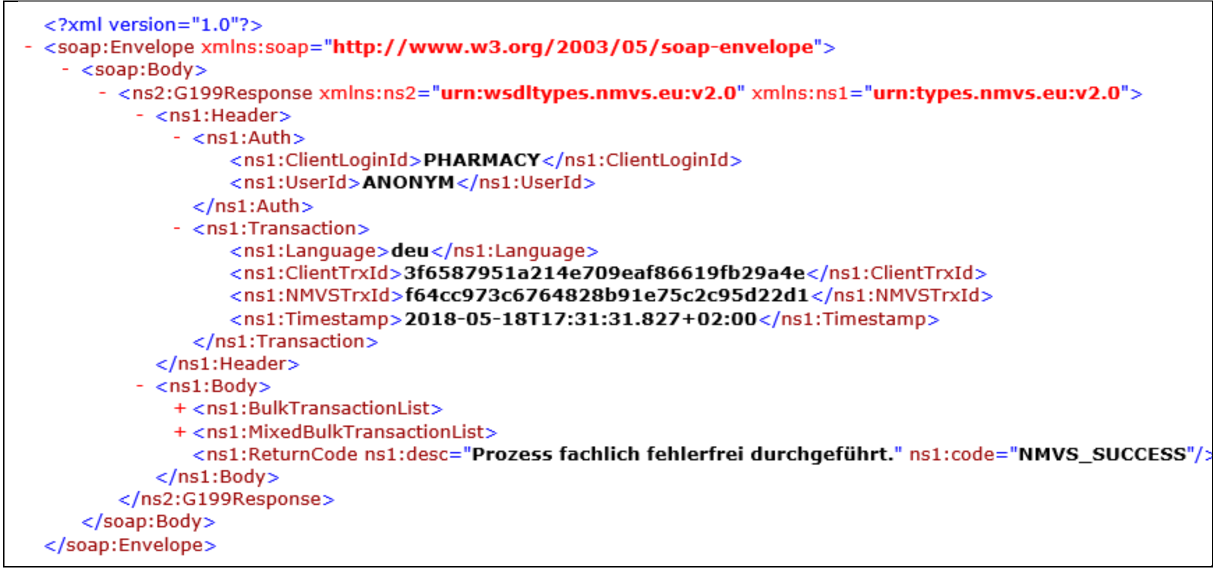
15.3.3 G199 - Pickup IDs for Bulk Result – Response Format

Attribute	O / M	Description
B – Body Data		
A3 – Transaction Data		
Client transaction ID	M	Identifies client transaction, e.g. dispense transaction coming from a pharmacy or wholesaler. Important for undo and error analysis.

Attribute	O / M	Description
NMVS transaction ID	M	Unique identifier of the transaction from the NMVS System.
B6 – Return Code		
Return code	M	Alphanumeric return code, e.g. NMVS_OK_03 or NMVS_ERR_21
Return code description	O	Textual description of the return code, e.g. password changed successfully or combination client/user/password invalid

Table 30: Body – Bulk Result Transaction Response Format

Example:



```
  <?xml version="1.0"?>
- <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
  - <soap:Body>
    - <ns2:G199Response xmlns:ns2="urn:wsdltypes.nmvs.eu:v2.0" xmlns:ns1="urn:types.nmvs.eu:v2.0">
      - <ns1:Header>
        - <ns1:Auth>
            <ns1:ClientLoginId>PHARMACY</ns1:ClientLoginId>
            <ns1:UserId>ANONYM</ns1:UserId>
          </ns1:Auth>
        - <ns1:Transaction>
            <ns1:Language>deu</ns1:Language>
            <ns1:ClientTrxId>3f6587951a214e709eaf86619fb29a4e</ns1:ClientTrxId>
            <ns1:NMVSTrxId>f64cc973c6764828b91e75c2c95d22d1</ns1:NMVSTrxId>
            <ns1:Timestamp>2018-05-18T17:31:31.827+02:00</ns1:Timestamp>
          </ns1:Transaction>
        </ns1:Header>
      - <ns1:Body>
        + <ns1:BulkTransactionList>
        + <ns1:MixedBulkTransactionList>
          <ns1:ReturnCode ns1:desc="Prozess fachlich fehlerfrei durchgeführt." ns1:code="NMVS_SUCCESS"/>
        </ns1:Body>
      </ns2:G199Response>
    </soap:Body>
  </soap:Envelope>
```

15.3.4 G199 - Use Case Diagram

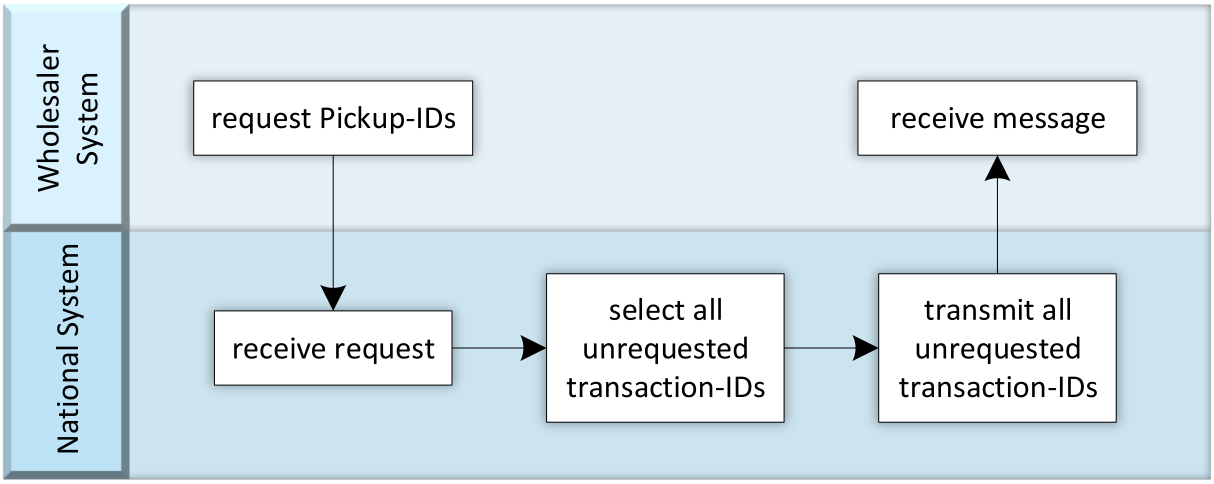


Figure 40: **Request Pickup IDs for Bulk Transaction Result**

Change Password from Pharmacy / Wholesaler

.

If, for any reasons, the password is lost or the user account gets locked (*NMVS _FE_AU_02*), it is possible to initiate account recovery in the NMVS Portal.

15.4 Password Expiry

End user passwords do have an expiry date. The initial password´s expiry date is 60 days after the creation. After that it depends on the environment.

15.4.1 Sandbox

After changing the initial password the new one will expire after 90 days.

15.4.2 IQE and PROD

On the IQE and the production environment the days until expiry can be configured during the installation of the NMVS. So the value depends on the NMVOs to define prior to the system setup.

15.4.3 End User Notification

The Notification is an optional field in the webservice responses, which provides additional information to the end user in certain circumstances. For example, prior to the expiration of the password, the NMVS will send out notifications with each answer via webservice. The notification will start X days before the expiration where X is to be defined by the NMVO and can only be changed with each installation of the NMVS.

15.5 Client Certificate

Besides the user/password authentication, NMVS uses client certificates as an additional security layer to legitimize user actions.

If no certificate is provided by a local trust store, HTTP Error 403 will be returned.

Installed but expired or withdrawn certificates will also result in HTTP 403.

As soon as a user access is created for a client system, the NMVS will automatically create and deliver an X.509-certificate for this user.

Not later than 30 days before the expiry date of a valid certificate, a new certificate will be generated. Delivery takes place via email containing a one-time TAN to acknowledge the delivery.

Retrieval can take place via the NMVS frontend (download) or via SOAP request. In both cases, the TAN will approve validity of the request.

The NMVS provides a response consisting of a passphrase protected PKCS#12 container holding the user certificate, the user's key-pair as well as the root CA.

In case of certificate enrolment or certificate renewal due to expiry all client systems should provide functionality to download and install clients in a local trust store.

See 13.4.4 - G615 - Download Certificate – Request Format for further information and examples.

15.6 Cipher Suites

See 8.2.1 HTTPS for a complete list of supported algorithm combinations.

15.7 PKI

These are the steps provided by NMVS public key infrastructure (PKI):

- NMVS provides a Certificate Authority
- NMVS generates a private/public key-pair for a user.
- NMVS enrols a X.509 user-certificate
- NMVS temporarily stores key-pair and certificate in an encrypted PKCS#12 container.
- NMVS keeps track of validity ranges and reminds users of certificate expiry.
- NMVS removes PKCS#12 container as soon as it is delivered to the user.

16 Permissions

A user must own the needed permissions for that particular service to access the API which means to issue calls and receive responses. Client applications can only request operations to which they have access via the permissions of the authenticated user.

The NMVO operating the API is in charge of authorising and restricting the availability of various features by configuring profiles and permission sets, and assigning users to them.

17 Sample Code

NMVS API works with current SOAP development environments, including, but not limited to Microsoft Visual Studio and Eclipse IDE. The Java examples will be based on JDK 8 as the C#.NET examples on C# 6. Examples for both languages were provided alongside this document.

17.1 C#

This section describes the process of creating a C# project to access the NMVS API using Visual Studio (minimum Version 2012, because .Net Framework 4.5 is needed), it is however transferable to other IDEs if necessary.

17.1.1 Create a New Project

Choose “File” → “New” → “Project” and choose “Visual C# Console application” from the list of templates. Type in the name of the application and the location and click “ok”.

Now you can start to develop the solution.

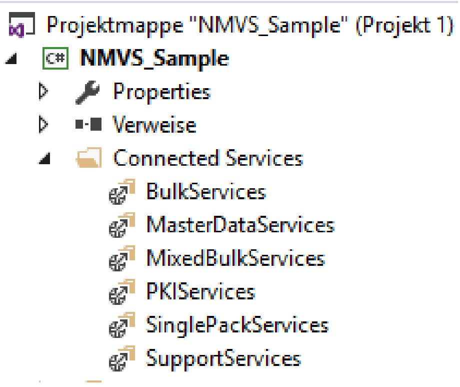
17.1.2 Generate Web Service Code

In this step, we will configure our project to automatically generate the classes needed to access the web services. First, copy the WSDL and XSD files into the project folder.

Include the WSDL files for the web services by right-click the name of the project in the solution explorer and choose “add service reference”. The Add Service Reference dialog box appears. Add the local path to your wsdl files in the address bar. For example: C:\WebServices\wsdl\WS_SINGLE_PACK.wsdl. Enter a meaningful name under namespace. For example SinglePackServices.

Use the following WSDL files: -WS_BULK -WS_Master_Data -WS_MIXED_BULK -WS_PKI -WS_SINGLE_PACK -WS_SUPPORT.

The project’s folder structure should look like this.

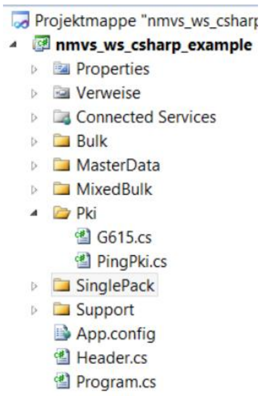


```
Projektmappe "NMVS_Sample" (Projekt 1)
◢ NMVS_Sample
    ▷ Properties
    ▷ Verweise
    ◢ Connected Services
        BulkServices
        MasterDataServices
        MixedBulkServices
        PKIServices
        SinglePackServices
        SupportServices
```

17.1.3 Invoking a Web Method

You can create classes somewhere in the application. In the example for every web service a folder is created and for every web service method a separate class was created.

With the main() method you can access to the different classes:



Projektmappe "nmvs_ws_csharp
- nmvs_ws_csharp_example
  - Properties
  - Verweise
  - Connected Services
  - Bulk
  - MasterData
  - MixedBulk
  - Pki
    - G615.cs
    - PingPki.cs
  - SinglePack
  - Support
  - App.config
  - Header.cs
  - Program.cs

17.1.4 Set up Client Certificate

We’ll need to provide a client certificate to authenticate ourselves to the NMVS server to use the other web services. For testing purposes, we can use the *UNMStestUSER.p12* file which contains a certificate recognized by the web service as well as a key pair. The file is provided alongside this documentation.

We need to set a reference to System.Security.Cryptography.X509Certificates to use this certificate file.

With following source code, you can include the certificate:

```
X509Certificate2 cert = new X509Certificate2(Properties.Settings.Default.CertificateFilePath
                       , Properties.Settings.Default.PrivateKeyPassword, X509KeyStorageFlags.PersistKeySet);
service.ClientCredentials.ClientCertificate.Certificate = cert;
```

The absolute path of the certificate is saved in the properties of the project.

17.2 Java

This section describes the process of creating a Java project to access the NMVS API using Apache Maven and Eclipse IDE (Oxygen)It will be, however, transferable to other IDEs if necessary.

17.2.1 Create a New Project

Choose “File” → “New” → “Project” and choose “Maven Project” from the list of wizards. Check “Create a simple project” and click “Next”. Enter your Group Id and an Artifact ID and click “Finish”.

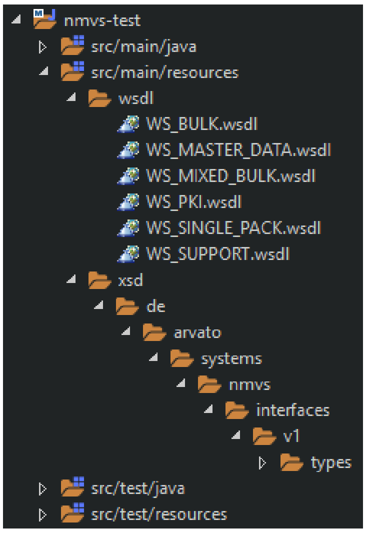
In your workspace, you should now see a new project with the Artifact ID as its name. You might need to change the compiler compliance level to allow current Java features to be used in your project. Right click the project and choose “Properties”. Click on “Java Compiler” in the navigation tree, uncheck “Use compliance from execution environment […]” and choose the highest level available from the dropdown for “Compiler compliance level” (1.8 at the time of writing).

The Java version used should also be specified in the POM: Open the pom.xml under the project’s root directory and add the following plugin declaration:

```
<project>
  ...
  <build>
    <plugins>
      <plugin>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.6.1</version>
        <configuration>
          <source>1.8</source>
          <target>1.8</target>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

17.2.2 Generate Web Service Code

In this step, we will configure our project to automatically generate the classes needed to access the web services. First, copy the WSDL and XSD files into the project folder under *src/main/resources*. The project's folder structure should look like pictured in Figure 42.



```
◢ nmvs-test
  ▷ src/main/java
  ◢ src/main/resources
    ◢ wsdl
        WS_BULK.wsdl
        WS_MASTER_DATA.wsdl
        WS_MIXED_BULK.wsdl
        WS_PKI.wsdl
        WS_SINGLE_PACK.wsdl
        WS_SUPPORT.wsdl
    ◢ xsd
      ◢ de
        ◢ arvato
          ◢ systems
            ◢ nmvs
              ◢ interfaces
                ◢ v1
                  ▷ types
  ▷ src/test/java
  ▷ src/test/resources
```

Figure 42: Project structure with the WSDL and XSD files.

Now we'll configure the *cxf-codegen-plugin* which is part of the Apache CXF services framework.

First, define a Maven property with the CXF version we'll use. At the time of writing, the latest version was 3.1.12:

```
<project>
  ...
  <properties>
    <cxf.version>3.1.12</cxf.version>
  </properties>
  ...
</project>
```

Next, declare the plugin in the POM's `<plugins>` section:

```
<project>
  ...
  <build>
    <plugins>
```

```
      <plugin>
        <groupId>org.apache.cxf</groupId>
        <artifactId>cxf-codegen-plugin</artifactId>
        <version>${cxf.version}</version>
        <executions>
          <execution>
            <id>generate-sources</id>
            <phase>generate-sources</phase>
            <goals>
              <goal>wsdl2java</goal>
            </goals>
            <configuration>
              <sourceRoot>${project.build.directory}/generated/cxf</sourceRoot>
              <wsdlRoot>${basedir}/src/main/resources/wsdl</wsdlRoot>
            </configuration>
          </execution>
        </executions>
      </plugin>
      ...
    </plugins>
    ...
  </build>
  ...
</project>
```

The above plugin declaration instructs Maven to invoke the plugin during the generate-sources build phase. The *cxf-codegen-plugin* will read the WSDLs and generate the necessary classes into *target/generated/cxf*. We need to declare three dependencies to be able to compile the generated code.:

```
<project>
  ...
  <dependencies>
    <dependency>
      <groupId>org.apache.cxf</groupId>
      <artifactId>cxf-rt-frontend-jaxws</artifactId>
      <version>${cxf.version}</version>
    </dependency>
    <dependency>
      <groupId>org.apache.cxf</groupId>
      <artifactId>cxf-rt-transports-http</artifactId>
      <version>${cxf.version}</version>
    </dependency>
    <dependency>
      <groupId>org.apache.cxf</groupId>
      <artifactId>cxf-rt-transports-http-jetty</artifactId>
      <version>${cxf.version}</version>
    </dependency>
    ...
  </dependencies>
  ...
</project>
```

Finally, add the generated sources to the Eclipse project's build path: In the project explorer, expand *target* and *generated*, right click on *cxf* and choose “Build Path” → “Use as Source Folder”.

17.2.3 Invoking a Web Method

The *cxf-codegen-plugin* generated a Java interface for each WSDL. The web service described in *WS_PKI.wsdl* does not require a client certificate, so we will try that one first.

Create a new class somewhere under *src/main/java* and create the `main()` method. There, we will obtain an instance of the web service interface, create a request object to pass to the ping method and print the response:

```
import eu.nmvs.services.v1_0.IPkiServices;
import eu.nmvs.services.v1_0.PkiServices;
import eu.nmvs.wsdltypes.v1.PkiPingRequest;
import eu.nmvs.wsdltypes.v1.PkiPingResponse;

public class Main {

    public static void main(String[] args) {
        // Get instance of WS_PKI service
        IPkiServices ws = new PkiServices().getPortPkiServices();

        // Set up request
        PkiPingRequest req = new PkiPingRequest();
        req.setInput("Hello World!");

        // Call web method
        PkiPingResponse res = ws.pingPki(req);

        // Print response
        System.out.println("Response: " + res.getOutput());
    }

}
```

17.2.4 Set up Client Certificate

We'll need to provide a client certificate to authenticate ourselves to the NMVS server to use the other web services. See chapter 15.4 for details on how to obtain such a certificate.

To use this certificate file, we need to set three system properties:

```
System.setProperty("javax.net.ssl.keyStore", "/PATH/TO/UNMStestUSER.p12");
System.setProperty("javax.net.ssl.keyStoreType", "pkcs12");
System.setProperty("javax.net.ssl.keyStorePassword", "PASSWORD");
```

These properties define the absolute path to the certificate file, the certificate type and the password to access it. Now, we can call methods from the other web services as well, for example:

```
// Get instance of WS_SINGLE service
ISinglePackServices ws = new SinglePackServices().getPortSinglePackServices();

// Set up request
G110Request req = new G110Request();
// fill request params
// req.setBody(...);

// Call web method
```

```
G110Response res = ws.g110Verify(req);

// Process response
// O1BodyType body = res.getBody();
// body.getProduct() ...
```

18 When to Call the NMVS

It is important that any software package calling the NMVS does so only when necessary. In one of its future versions, the NMVS will provide so called “Inter-Market Transactions”. An Inter-Market Transaction occurs when a EU medicament pack is scanned but the corresponding product has no market authorization for the domestic market. This happens for example when medicament packs from another EU market are imported to cover a temporary medicament shortage.

When such a locally unknown product pack is verified, the NMVS will launch a verification request towards the EU Hub which in turn will forward this request to other local NMVS’s to see whether the pack can be found there. This is a resource intensive process which should not be triggered unnecessarily.

The problem with this is that serialized non-medicaments may exist on the local market. As these products (e.g. syringes) are also unknown to the local NMVS no NMVS web service calls must be made for them.

The diagram below shows a product landscape and identifies those serialized products which have to be verified (dotted frames).

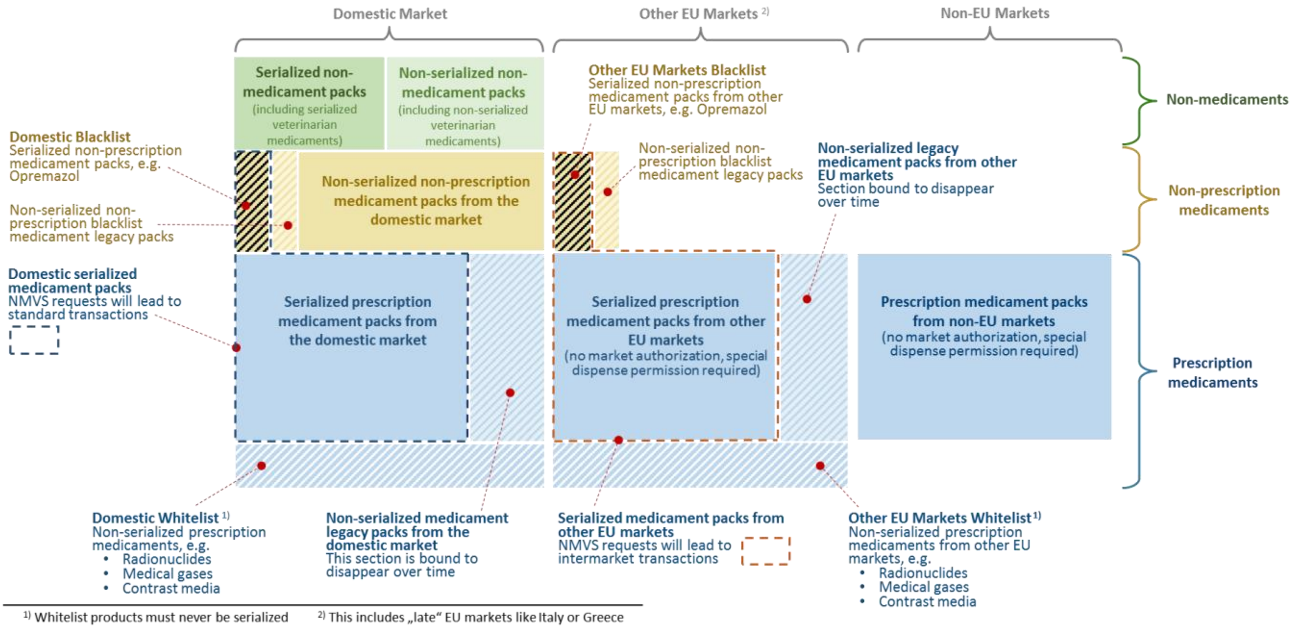
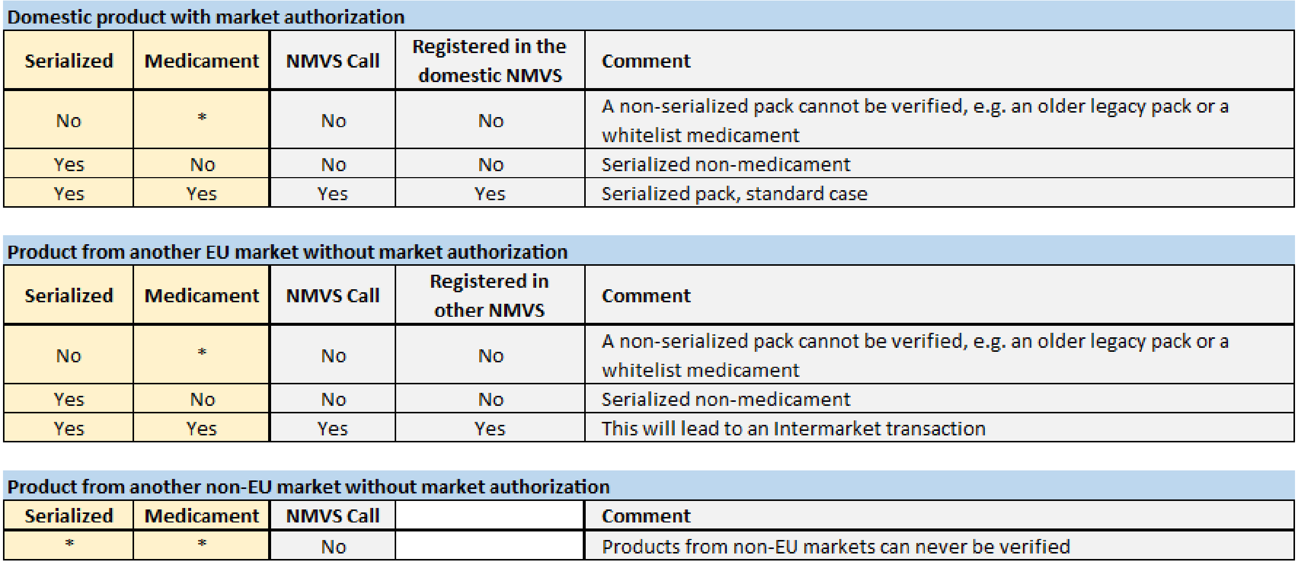

Domestic Market
Other EU Markets 2)
Non-EU Markets
Serialized non-medicament packs (including serialized veterinarian medicaments)
Non-serialized non-medicament packs (including non-serialized veterinarian medicaments)
Other EU Markets Blacklist
Serialized non-prescription medicament packs from other EU markets, e.g. Opremazol
Non-medicaments
Domestic Blacklist
Serialized non-prescription medicament packs, e.g. Opremazol
Non-serialized non-prescription blacklist medicament legacy packs
Non-serialized non-prescription medicament packs from the domestic market
Non-serialized non-prescription blacklist medicament legacy packs
Non-serialized legacy medicament packs from other EU markets
Section bound to disappear over time
Non-prescription medicaments
Domestic serialized medicament packs
NMVS requests will lead to standard transactions
Serialized prescription medicament packs from the domestic market
Serialized prescription medicament packs from other EU markets (no market authorization, special dispense permission required)
Prescription medicament packs from non-EU markets (no market authorization, special dispense permission required)
Prescription medicaments
Domestic Whitelist 1)
Non-serialized prescription medicaments, e.g.
• Radionuclides
• Medical gases
• Contrast media
Non-serialized medicament legacy packs from the domestic market
This section is bound to disappear over time
Serialized medicament packs from other EU markets
NMVS requests will lead to intermarket transactions
Other EU Markets Whitelist 1)
Non-serialized prescription medicaments from other EU markets, e.g.
• Radionuclides
• Medical gases
• Contrast media
1) Whitelist products must never be serialized
2) This includes „late“ EU markets like Italy or Greece

Figure 43: Serialized Product Landscape

The next two figures describe the criteria that can be used to decide when to call the NMVS and when not to call it.



Domestic product with market authorization

Serialized	Medicament	NMVS Call	Registered in the domestic NMVS	Comment
No	*	No	No	A non-serialized pack cannot be verified, e.g. an older legacy pack or a whitelist medicament
Yes	No	No	No	Serialized non-medicament
Yes	Yes	Yes	Yes	Serialized pack, standard case

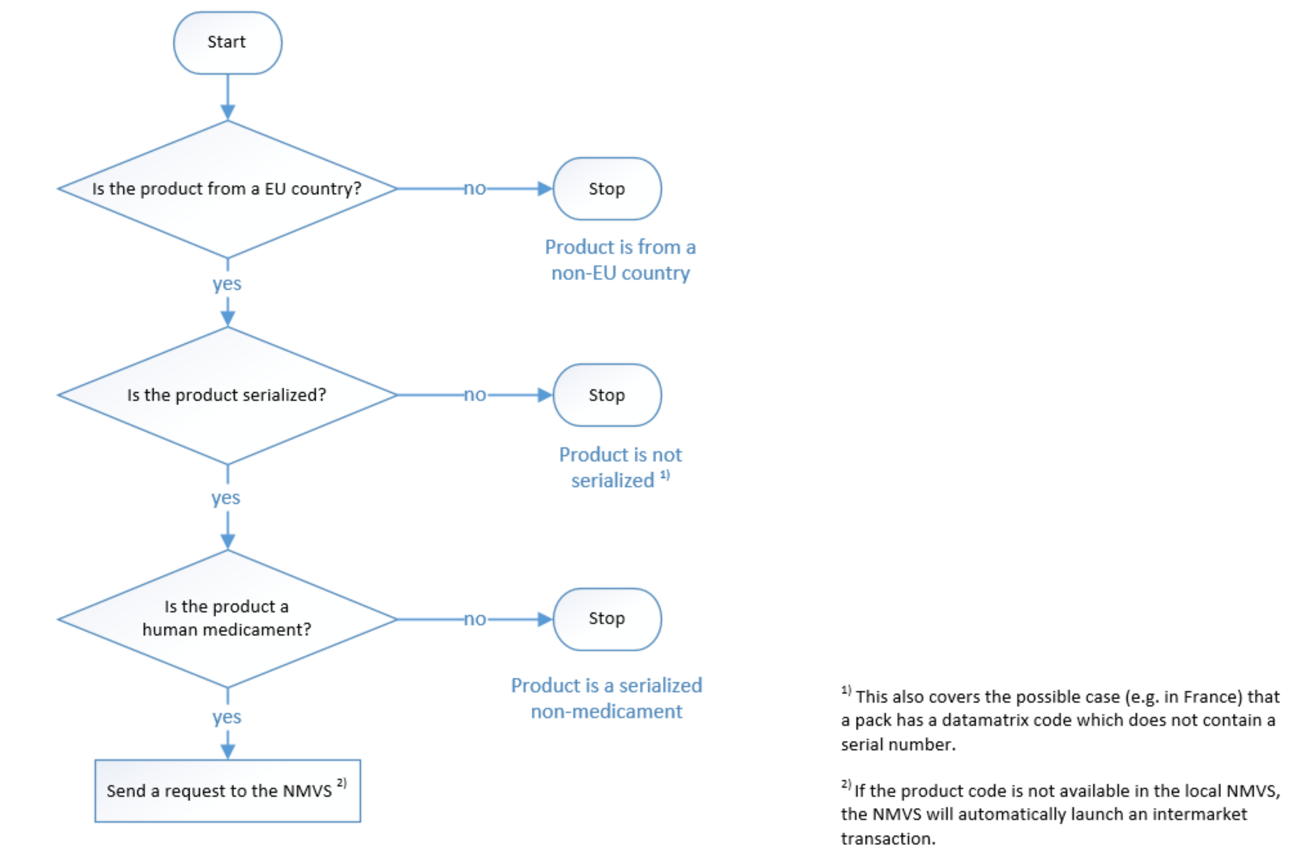
Product from another EU market without market authorization

Serialized	Medicament	NMVS Call	Registered in other NMVS	Comment
No	*	No	No	A non-serialized pack cannot be verified, e.g. an older legacy pack or a whitelist medicament
Yes	No	No	No	Serialized non-medicament
Yes	Yes	Yes	Yes	This will lead to an Intermarket transaction

Product from another non-EU market without market authorization

Serialized	Medicament	NMVS Call		Comment
*	*	No		Products from non-EU markets can never be verified

Figure 44: Decision Tables for NMVS Calls


Start
Is the product from a EU country?
no
Stop
Product is from a non-EU country
yes
Is the product serialized?
no
Stop
Product is not serialized 1)
yes
Is the product a human medicament?
no
Stop
Product is a serialized non-medicament
yes
Send a request to the NMVS 2)

[1] This also covers the possible case (e.g. in France) that a pack has a datamatrix code which does not contain a serial number.

[2] If the product code is not available in the local NMVS, the NMVS will automatically launch an intermarket transaction.

Figure 45: Decision Diagram for NMVS Calls

19 Data Matrix Encoding Standards

While all data matrix codes for serialized EU medicaments are based on the same technical standard (https://en.wikipedia.org/wiki/Data_Matrix), the content structure of the codes may differ.

The NMVS supports two different product number coding schemes, *GTIN* and *PPN*. GTIN encoded product numbers are very common all over Europe whereas PPNs are mainly used in Germany.

However, in the context of Intermarket Transactions, PPN encoded product packs may appear in other countries and there will be an obligation to verify them on dispense. Software suppliers should therefore prepare for this possibility.

20 REST Interface

Note: Please note that the REST interface is only intended for the **NMVS in Spain.** If you support another country, please continue to use the SOAP interfaces.

The NMVS software is upon request configurable to provide a REST (Restful State Transfer) based interface. Please make sure to find out with your local NMVO which interface to use. In most countries the SOAP interface will be used.

The chapter at hand describes the NMVS REST Interface on a more business related view than on a technical view. It has to be seen as addendum to the previous chapters covering the SOAP interface.

The capability to provide REST only affects the interface, not the internal processing within the NMVS. That means all business use cases and mandatory and optional parameters in requests and answers remain identical to the ones described in previous chapters that deal with the SOAP interface. For that reason even if you are going to use the REST interface, you should be familiar with the SOAP interface and the request and answer format.

Again, the REST or the SOAP interface will only be available on request of the local NMVO. Make sure to use the right interface!

20.1 General Request / Response Structure

Similar to the SOAP interface format, all requests and responses consist of a header and a body.

To avoid redundant pieces of information between documents we only show an example of request and response in “Figure 46: Example General Request” and “Figure 47: Example General Response”

```
{
   "header": {
      "auth": {
         "clientLoginId": "G1234-12-12RT",
         "userId": "1234",
         "password": "otgPGUI76#",
         "subUserId": "4711"
      },
      "userSoftware": {
         "name": "EasyPharm",
         "supplier": "PharmSoft Ltd.",
         "version": "3.2.0"
       },
       "transaction": {
         "clientTrxId": "c295a450-0caa-4cf1-8e62-c6165c4",
         "language": "deu"
       }
     },
     "body": {
       "product": {
         "productCode": {
           "scheme": "GTIN",
           "value": "05060141900015"
         },
         "batch": {
           "id": "1264",
           "expDate": "171212"
         }
       },
       "pack": {
         "sn": "XVTR75973491006155"
       }
     }
}
```

Figure 46: Example General Request

```
{
   "header":    {
      "auth":       {
         "clientLoginId": "G1234-12-12RT",
         "userId": "1234",
         "subUserId": "4711"
      },
      "transaction":       {
         "language": "deu",
         "clientTrxId": "c295a450-0caa-4cf1-8e62-c6165c4",
         "timestamp": 1516186576798,
         "nmvstrxId": "c295a450-0caa-4cf1-7e62-c6865d41"
      }
   },
   "body":    {
      "product":       {
         "productCode":          {
            "value": "05060141900015",
            "scheme": "GTIN"
         },
         "intermarketFlag": true,
         "nhrn": "09556886",
         "productInfoRef": null,
         "batch":          {
            "id": "1264",
           "expDate": "171231",
            "expDateNMVS": 1542927600000
         }
      },
      "pack":       {
         "reason": ["SUPPLIED"],
         "sn": "XVTR75973491006155",
         "state": "INACTIVE"
      },
      "returnCode":       {
         "code": "NMVS_SUCCESS",
         "desc": "Successfully processed"
      }
   }
}
```

Figure 47: Example General Response

20.1.1 Content (Header, Body, Errors)

The header contains the authentication information, information related to the software that requests the NMVS service and transaction IDs for identifying the transaction.

The body of a typical request contains all input data necessary for the processing such as product, batch and pack.

The body of response contains the processing result. In some cases like errors, it may only contain a return code, in other cases it will be the input data, additional data from NMVS, the result of processing and a return code.

For detailed information concerning the request- and response structure, please read the corresponding chapter "General Request/Response Structure".

The general errors are divided in two parts, on the one hand parser errors and on the other hand authentication errors and errors that prevent processing. If the NMVS is not able to read the request because of JSON errors, the NMVS will send a fault message. In this case the fault string contains the error code and the error message. Example:

```
{
   "returnCodeMessage": "Input data does not match the JSON schema definition.",
   "exceptionMessage": "cvc-pattern-valid: Value '?' is not facet-valid with respect to pat-
                        tern '([A-Z]*[a-z]*\\d*-*)+' for type 'ClientLoginId_Type'.",
   "backendReturncode": "NMVS_TE_JS_02"
}
```

Figure 48: Example Error Response - JSON Schema Incorrect

In other cases, if the authentication fails or the NMVS gets an error during processing the data, an error with the normal response header and return code will be passed back:

```
{
   "header":    {
      "auth":       {
         "clientLoginId": "G1234-12-12RT",
         "userId": "1234",
         "subUserId": "4711"
      },
      "transaction":       {
         "language": "deu",
         "clientTrxId": "c295a450-0caa-4cf1-8e62-c6165c4",
         "timestamp": 1516186576798,
         "nmvstrxId": "c295a450-0caa-4cf1-7e62-c6865d41"
      }
   },
   "body": {
      "returnCode":    {
         "code": "NMVS_FE_AU_01",
         "desc": "The entered user or password does not match."
      }
   }
}
```

Figure 49: Example Error Response - During Processing the Data

20.2 Synchronous Web Service Operations

For detailed information and description concerning any web service operation, please read the corresponding chapter "Synchronous Web Service Operations".

20.2.1 Single Pack Transactions

20.2.1.1 Do Single Pack Transactions

20.2.1.1.1 Request Format

```
        "body": {
           "product": {
             "productCode": {
               "scheme": "PPN",
               "value": "110258282253"
             },
             "batch": {
               "id": "U0226",
               "expDate": "121217"
             }
           },
           "pack": {
             "sn": "E0XG32FRM57GGSJZ4AMG"
           }
         }
```

Figure 50: Example Request Format - Single Pack Transaction

The request for **manual entry** is exactly the same, except that the batch data is not needed:

```
        "body": {
           "product": {
             "productCode": {
               "scheme": "PPN",
               "value": "110258282253"
             }
           },
           "pack": {
             "sn": "E0XG32FRM57GGSJZ4AMG"
           }
         }
```

Figure 51: Example Request Format - Single Pack Transaction Manual Entry

Attribute	O / M	Description
For a description of parameters read the chapters “Single Pack Transaction – Request Format” and “Single Pack Transaction – Request Format for Manual Data Entry”.		

20.2.1.1.2 Response Format

```
    "body": {
       "product": {
          "productCode": {
             "value": "05060141900015",
             "scheme": "GTIN"
          },
          "intermarketFlag": true,
          "nhrn": "09556886",
          "productInfoRef": null,
          "batch": {
             "id": "1264",
            "expDate": "171231",
             "expDateNMVS": 1542927600000
```

```
         }
      },
      "pack": {
         "reason": ["SUPPLIED"],
         "sn": "XVTR75973491006155",
         "state": "INACTIVE"
      },
      "returnCode": {
         "code": "NMVS_SUCCESS",
         "desc": "Successfully processed"
      }
   }
```

Figure 52: Example Response Format - Single Pack Transaction

Attribute	O / M	Description
For a description of parameters read the chapter “Single Pack Transaction – Response Format”		

20.2.1.2 Undo Single Pack Transactions

20.2.1.2.1 Request Format

```
   "body": {
      "product": {
         "productCode": {
            "scheme": "PPN",
            "value": "110258282253"
         },
         "batch": {
            "id": "U0226",
            "expDate": "121217"
         }
      },
      "pack": {
         "sn": "E0XG32FRM57GGSJZ4AMG"
      },
      "refClientTrxId": "c295a450-0caa-4cf1-8e62-c6165c3"
   }
```

Figure 53: Example Request Format - Undo Single Pack Transaction

Attribute	O / M	Description
For a description of parameters read the chapters “Undo Single Pack Transaction – Request Format”		

20.2.1.2.2 Response Format

The response format of an undo single pack transaction is exactly the same as the response of a do single pack transaction (chapter 20.2.1.1.2).

20.3 Asynchronous Web Service Operations

Bulk transaction describes the service to be used by a distributor to handle a bigger number of packs with the same product code.

For detailed information and description concerning any web service operation, please read the corresponding Chapter “Asynchronous Web Service Operations – Homogenous Bulk Transactions”.

20.3.1 Homogenous Bulk Transactions

20.3.1.1 Do Bulk Transactions

20.3.1.1.1 Request format

```
    "body": {
      "product": {
        "productCode": {
          "scheme": "GTIN",
          "value": "05060141900015"
        },
        "batch": {
          "id": "05060141900015",
          "expDate": "171212"
        }
      },
      "packs": {
        "pack": [{
          "sn": "XVTR75973491006155",
          "sn": "XVTR75973491006156",
          "sn": "XVTR75973491006157",
          "sn": "XVTR75973491006158"
         }]
      }
    }
```

Figure 54: Example Request Format - Bulk Transaction

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction – Request Format”		

20.3.1.2 Undo Bulk Transactions

20.3.1.2.1 Request Format

```
    "body": {
      "product": {
        "productCode": {
          "scheme": "GTIN",
          "value": "05060141900015"
        },
        "batch": {
          "id": "05060141900015",
          "expDate": "171212"
        }
      },
      "packs": {
        "pack": [{
          "sn": "XVTR75973491006155",
          "sn": "XVTR75973491006156",
          "sn": "XVTR75973491006157",
          "sn": "XVTR75973491006158"
        }]
      },
      "refClientTrxId": "c295a450-0caa-4cf1-8e62-c6165c3"
    }
```

Figure 55: Example Request Format - Undo Bulk Transaction

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction Use Cases”		

20.3.1.3 G188 - Request Bulk Pack Operation Result

20.3.1.3.1 Request Format

```
    "body": {
      "refClientTrxId": "c295a450-0caa-4cf1-8e62-c6165c3"
    }
```

Figure 56: Example Request Format - Request Bulk Pack Operation Result

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction Use Cases”		

20.3.1.3.2 Response Format

```
    "body": {
       "message": null,
       "product": {
          "productCode": {
             "value": "110258282253",
             "scheme": "PPN"
          },
          "intermarketFlag": null,
          "nhrn": "02582822",
          "productInfoRef": null,
          "batch": {
             "id": "U0226",
             "expDate": "190919",
             "expDateNMVS": 1568844000000
          }
       },
       "packs": {"pack":       [{
             "reason": [],
             "sn": "E0XG32FRM57GGSJZ4AMG",
             "state": "ACTIVE",
             "returnCode":             {
                "code": "NMVS_SUCCESS",
                "desc": "Successfully processed"
             }
          },
                   {
             "reason": [],
             "sn": "Z4165HUZGUOH466JIUHP",
             "state": "UNKNOWN",
             "returnCode":             {
                "code": "NMVS_NC_PC_02",
                "desc": "Unknown serial number."
             }
          }
       ]},
       "returnCode":       {
          "code": "NMVS_SUCCESS",
          "desc": "Successfully processed"
       }
    }
```

Figure 57: Example Response Format - Request Bulk Pack Operation Result

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction Use Cases”		

20.3.1.4 G199 - Request Pickup IDs for Bulk Transaction Result

20.3.1.4.1 Request Format

The request of this process does not contain a body. The transaction ID in the header of the request is sufficient to execute this process.

```
"transaction": {
  "clientTrxId": "c295a450-0caa-4cf1-8e62-c6165c4",
  "language": "deu"
}
```

Figure 58: Example Request Format - Request Pickup IDs for Bulk Transaction Result

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction Use Cases”		

20.3.1.4.2 Response Format

```
    "body": {
      "bulkTransactionList": {"transaction": []},
      "mixedBulkTransactionList": {"transaction": []},
      "returnCode":{
        "code": "NMVS_SUCCESS",
        "desc": "Successfully processed"
      }
    }
```

Figure 59: Example Response Format - Request Pickup IDs for Bulk Transaction Result

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction Use Cases”		

20.3.2 Mixed Bulk Transactions

20.3.2.1 G195 - Submit Mixed Bulk Transaction

20.3.2.1.1 Request Format

```
    "body": {
       "trxList": {
          "trxItem": [{
             "item": {
                "reqType": "G_110",
                "product": {
                   "productCode": {
                      "scheme": "PPN",
                      "value": "110258282253"
                   },
                   "batch": {
                      "id": "U0226",
                      "expDate": "171212"
                   }
                },
                "pack": {"sn": "E0XG32FRM57GGSJZ4AMG"},
                "transaction": {
                   "clientTrxId": "1234567",
                   "language": "deu",
                   "refClientTrxId": "123456"
                }
             }
          }]
       }
    }
```

Figure 60: Example Request Format - Submit Mixed Bulk Transaction

Attribute	O / M	Description
For a description of parameters read the chapter “Homogenous Bulk Transaction Use Cases”		

20.3.2.2 G196 - Request Mixed Bulk Transaction Result

20.3.2.2.1 Request Format

```
    "body": {
      "refClientTrxId": "ab7e1d8893c44443a429c6f121f091e6"
    }
```

Figure 61: Example Request Format - Request Mixed Bulk Transaction Result

Attribute	O / M	Description
For a description of parameters read the chapter “Mixed Bulk Transaction Use Cases”		

20.3.2.2.2 Response Format

```
   "body": {
      "trxList": {
         "trxItem": [{
            "transaction": {
               "language": "deu",
               "clientTrxId": "1234567",
               "timestamp": 1516636604431,
               "nmvstrxId": "245e28bf5a3a45f38b18be32142765d3"
            },
            "product": {
               "productCode": {
                  "value": "110258282253",
                  "scheme": "PPN"
               },
               "intermarketFlag": null,
               "nhrn": "02582822",
               "productInfoRef": null,
               "batch": {
                  "id": "U0226",
                  "expDate": "171212",
                  "expDateNMVS": null
               }
            },
            "pack": {
               "reason": [],
               "sn": "E0XG32FRM57GGSJZ4AMG",
               "state": "UNKNOWN",
               "returnCode": {
                  "code": "NMVS_FE_LOT_03",
                  "desc": "Selected batch designation does not exist."
               }
            },
            "reqType": "G_110"
         }]
      },
      "returnCode": {
         "code": "NMVS_SUCCESS",
         "desc": "Successfully processed"
      }
```

Figure 62: Example Response Format - Request Mixed Bulk Transaction Result

Attribute	O / M	Description
For a description of parameters read the chapter “Mixed Bulk Transaction Use Cases”		

20.4 Support Use Cases

20.4.1 G445 - Change Password

20.4.1.1 Request Format

```
    "body": {
      "password": "O4ub!uPz",
      "newPassword": "Fa5eh5sj$"
    }
```

Figure 63: Example Request Format - Change Password

Attribute	O / M	Description
For a description of parameters read the chapter “G445 Change Password - Request Format”		

20.4.1.2 Response Format

```
    "body": {
      "returnCode": {
         "code": "NMVS_SUCCESS",
         "desc": "Successfully processed"
      }
    }
```

Figure 64: Example Response Format - Change Password

Attribute	O / M	Description
For a description of parameters read the chapter “G445 Change Password - Response Format”		

20.5 PKI Use Cases

20.5.1 G615 - Download Certificate

20.5.1.1 Request Format

```
    "body": {
      "tan": "126334"
    }
```

Figure 65: Example Request Format - Download Certificate

<table>
<tr><th>Attribute</th><th>O / M</th><th>Description</th></tr>
<tr><td colspan="3">For a description of parameters read the chapter “G615 Download Certificate - Request Format”</td></tr>
</table>

20.5.1.2 Response Format

```
    "body": {
       "cert": {
          "value": "MIIKbgIBAzCCCigGCSqGSIb3DQEHAaCCChkEggoVMIIK==",
          "name": "MAX",
          "passphrase": "5QBQBfPt"
       },
       "returnCode": {
          "code": "NMVS_SUCCESS",
          "desc": "Successfully processed"
       }
    }
```

Figure 66: Example Response Format - Download Certificate

<table>
<tr><th>Attribute</th><th>O / M</th><th>Description</th></tr>
<tr><td colspan="3">For a description of parameters read the chapter “G615 Download Certificate - Response Format”</td></tr>
</table>