WWKS 2

Reference Manual

P-010-044-B-I-599-GBR.docx-V6.0

WWKS 2

Translation of the original reference manual

Updates

Updates to hardware, software, and methods are made continuously; related information will be included in future versions of the manual. Make sure that you are working with the latest version of the manual.

Manufacturer

Becton Dickinson Rowa Germany GmbH, Rowastraße, 53539 Kelberg, Germany
Phone: +49-2692-9206-0, Fax: +49-2692-9206-1299
www.bd.com/rowa, rowa@bd.com

In the following, "Rowa" always refers to the manufacturer, Becton Dickinson Rowa Germany GmbH. Becton Dickinson Rowa Germany GmbH is part of Becton, Dickinson and Company headquartered in Franklin Lakes, USA.

Service

Please contact your sales/service partner or Rowa directly:

- Toll-free within Germany: 0800-2273-3874-66
- Toll call outside Germany: 0700-5154-5758
- Email: rowa-service@bd.com

Important information

- This instruction manual is a component of the product. The instruction manual should be available to users of the product at all times.
- Read the instruction manual carefully before using the product: Pay attention to all warnings and notes in order to operate the product safely, correctly, and reliably.

Liability limitations

The manufacturer assumes no liability for damage caused by the following:

- Inappropriate or improper use
- Subsequent modifications made by the operator or a third party
- Non-compliance with the instruction manual
- Use by untrained personnel

Contents

2 Introduction

What is this document about?

This document details the communication between a pharmacy IT system and an automated storage and retrieval system using the TCP/IP protocol. The exchanged messages are UTF-8 coded XML messages. The message content is embedded in tags based on the WWKS 2 standard.

The messages are exchanged in both directions, and are used to control inventory movements (stock input and output) in an automated storage and retrieval system (picking system).

This manual sets out the available commands, their usage and syntax. With this information, programmers are able to assure successful communication between their pharmacy IT system and an automated storage and retrieval system via WWKS 2.

Who should read this document?

This document is intended for

- personnel involved in interfacing an automated storage and retrieval system to a pharmacy IT system.
- programmers developing software (such as parsers) to translate between WWKS 2 messages and the pharmacy IT system.

Terminology

The two systems are abbreviated as follows in places within the text:

- Pharmacy IT system = WWS
- Automated storage and retrieval system = KS
  In the context of this document, an automated storage and retrieval system also means a system comprising multiple interconnected automatic storage machines with or without a shared control computer.

3 Pharmacy IT system and automated storage and retrieval system

Operators of automated storage and retrieval systems have a pharmacy IT system by which they build, manage and ship their drug inventories. The inventory management system generates the requests for the automated storage and retrieval system to process.

Typical application cases are:

- Recognition and inputting of medication packs
- Issue of drugs from stock to pharmacy staff when a customer presents a prescription
- Printing of pack labels for individual patients

This is just a small selection of the very many conceivable actions which require the assignment of current medication demand and physical pack transport.

Purely informative data traffic also occurs in practice, e.g.

- Querying of stock levels

As there are many different inventory management systems, it is advisable to operate standards such as HL7, WWKS or WWKS 2. WWKS 2 has established itself as a powerful and future-proof interface. In hospitals in particular, where large orders, batch and serial numbers, pack sizes, opened packs and data protection frequently play a role, WWKS 2 is able to satisfy the diverse information and regulatory requirements.

Communication between the pharmacy IT system and the automated storage and retrieval system is run by the TCP/IP protocol. The operator's network and the automated storage and retrieval system are connected via a router. In most cases the automated storage and retrieval system acts as the server and the pharmacy IT system as the client. The inventory management system connects via port 6050 on the router (in the case of multiple systems via port 6053). The IP address of the router is either assigned via the DHCP or configured as a fixed address of the operator's network.

The automated storage and retrieval system waits for incoming requests from the pharmacy IT system. The data link is made only by the pharmacy IT system and should remain active as long as the automated storage and retrieval system and the pharmacy IT system are in operation. The link is not opened and closed for each individual action. UTF8-coded XML messages are exchanged in both directions via the link.

The (inventory management system) operator's software specialist writes a software program which prepares the WWKS 2 messages for the inventory management system.

The programming commitment depends on each individual case, owing to the diversity of different inventory management systems in use. Work on the system can in fact begin before the automated storage and retrieval system has been physically constructed. An emulation program for WWKS 2 is available for testing of the communication between the pharmacy IT system and the automated storage and retrieval system.

To aid programming of the WWKS 2 interface, Rowa provides a .NET 4.0 based library as an API.

4 Structure of the manual

Breakdown

This manual assigns WWKS 2 messages to the following function groups:

- Initialization — Prepare message exchange between pharmacy IT system and automated storage and retrieval system
- Keepalive — Check data link
- System status — Poll automated storage and retrieval system readiness
- Stock input — Store packs in automated storage and retrieval system
- Stock input initiation — Trigger pack input into automated storage and retrieval system
- Master data — Transfer article and shipping lists
- Inventory checking — Check automated storage and retrieval system stock levels
- Stock output — Output packs from automated storage and retrieval system
- Task status — Poll task processing status
- Cancellation — Cancel tasks
- Configuration checking — Check configuration of automated storage and retrieval system
- Stock location checking — Check stock location information

Flowcharts

The flowcharts based on the UML2 standard depict the flow of messages between the pharmacy IT system and automated storage and retrieval system as well as the decisions and actions triggered as a result. If the sequence is dependent on conditions, the variants are depicted by branching.

Element tables

The message element tables describe the syntax and the possible values of the elements.

The "M/O" column indicates whether the element is mandatory (M) or optional (O).

If an element can be used more than once per message, the fact is mentioned in the description. If this attribute is not mentioned, the element can be used only once.

Examples

An example is presented for each message described. The example messages have fictitious values.

Library

The corresponding methods of the WWKS 2 library and its usage are presented for each message with examples in the C# language.

You will find general information on the library in section *WWKS 2 library*.

5 Data structures

5.1 Data types

The following data types are used in the messages:

Data type	Description
Tag	XML tag
String	Character string (corresponding to the data type *string* of the W3C specification for the XML scheme) The permissible value ranges are defined as follows: *#x9* | *#xA* | *[#x20-#xD7FF]* | *[#xE000-#xFFFD]* | *[#x10000-#x10FFFF]* and each unicode character apart from the surrogate blocks *FFFE* and *FFFF*, as specified in the standard ISO/IEC 10646. All control characters (e.g. *#x1D*) not specified in the ISO standard must be coded with *\x00*. The zeros represent the respective HEX code. Example: *\x1D.* These characters must be substituted: *&* by *&* *<* by *<* *>* by *>* *"* by *"* *'* by *'*
Integer 32-bit	32-bit integer (corresponding to the data type *int* of the W3C specification for the XML scheme) This data type has a syntax comprising a sequence of integers (*#x30-#x39*). Thousands separators and decimal points are not allowed. The permitted value range covers the numbers between -2147483648 and 2147483647.
Integer 64-bit	64-bit integer (corresponding to the data type *long* of the W3C specification for the XML scheme) This data type has a syntax comprising a sequence of integers (*#x30-#x39*). Thousands separators and decimal points are not allowed. The permitted value range covers the numbers between -9223372036854775808 and 9223372036854775807.
Boolean	One of the statements *True* or *False*

In some cases, the types described here are restricted if they are used in attributes (e.g. 32-bit integer > 0). This is noted at the relevant locations.

5.2 Message structure

Each message is embedded in the container element *WWKS*, which is constructed according to this scheme:

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
</WWKS>
```

The element *WWKS* encloses each individual message. All other elements of a message are subelements of *WWKS*.

The attribute *Version* currently has the value "2.0".

The attribute *TimeStamp* is a time stamp in extended UTC format (coordinated universal time).

In each message, *WWKS* is followed by the "lead" element, which determines the message type.

Example message:

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputRequest Id="1002" Source="999" Destination="100" IsNewDelivery="True">
    <Article>
      <Pack Index="0" DeliveryNumber="463526"
       ScanCode="0004-56-034-G|00007|T|Omepra0004|PalH09051200001|051112|XXXXXX"/>
    </Article>
  </InputRequest>
</WWKS>
```

In the example, the lead element *InputRequest* determines the message type: stock input request for a pack. *InputRequest* is followed by additional subelements further specifying the input request.

5.3 WWKS 2 library

To aid programming of the WWKS 2 interface, Rowa provides a library as an API.

The software consists of the following files:

- *log4net.dll* — Dependency of the file *StorageSystem.dll* to implement logging
- *StorageSystem.dll* — The WWKS 2 library
- *StorageSystem.dll.log4net* — Log configuration file

Include these files in your project.

The programming language you use must be compatible with the .NET 4.0 framework. The programming examples in this manual are written in C#, as is the library itself.

To enable customization to specific needs, the source code is provided along with the library.

General information on programming

The central point of access is the class `RowaStorageSystem`. This implements the `IStorageSystem` interface, which provides a complete interface to a automated storage and retrieval system.
The `RowaStorageSystem` class corresponds to the interface to a Rowa system.
Instancing is implemented via the `new`operator, while clearing is implemented by the `Dispose` method:

```
IStorageSystem storageSystem = new RowaStorageSystem();
storageSystem.Dispose();
```

As `IStorageSystem` inherits from `IDisposable`, the instruction `using` can also be used in:

```
using (IStorageSystem storageSystem = new RowaStorageSystem())
{
      // do your actions here
}
```

Use of the methods

The individual methods of the library are described in the *message reference* on the following pages. The methods are assigned to the XML messages to which they correspond. For each of the "lead" message elements described in the reference you will find a subsection headed *Library*, where the methods that support the desired message are listed.
Commentaried code examples demonstrate the possible procedure. Code add-ons from Visual Studio or other development environments will show the developer further elements and defined values.

The methods process the message pairs ...*Request* and ...*Response* mostly together.
Example: The `UpdateMasterArticles` method sends the *ArticleMasterSetRequest*, waits for the corresponding *ArticleMasterSetResponse* and evaluates it.

6 Message reference

6.1 Initialization

Lead elements

HelloRequest

HelloResponse

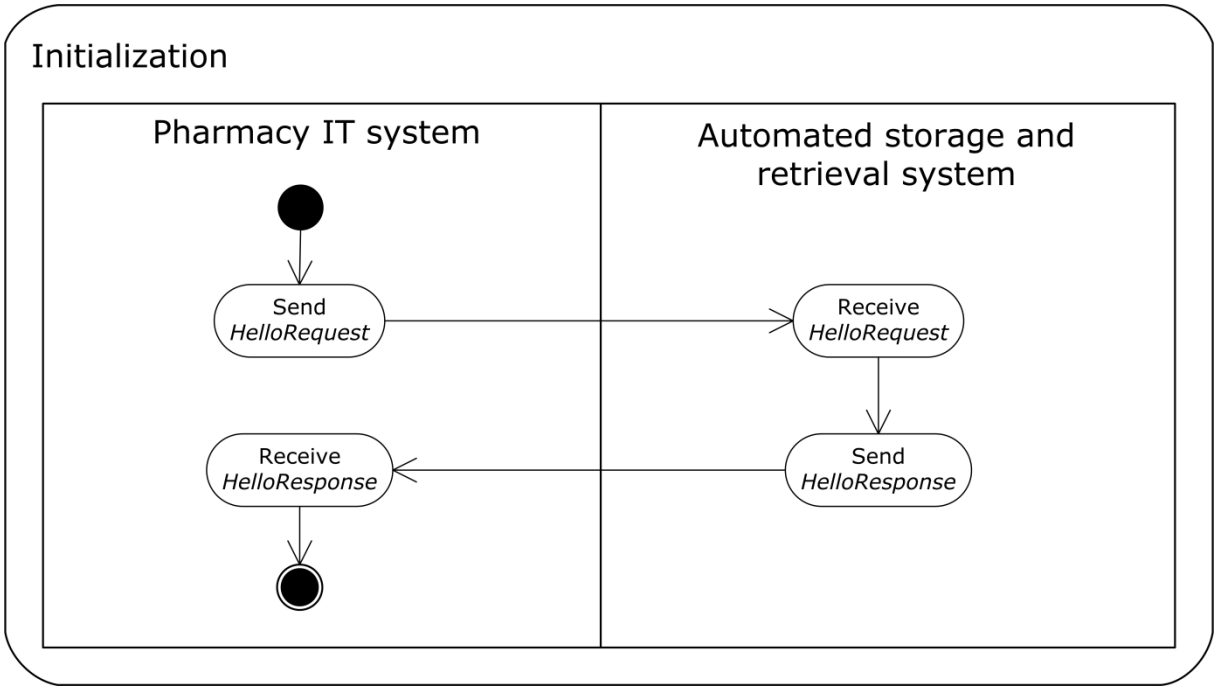
Usage

When the TCP/IP connection between the pharmacy IT system and the automated storage and retrieval system has been made, the pharmacy IT system sends the *HelloRequest* once to prepare for the further communications with the automated storage and retrieval system. The automated storage and retrieval system responds with *HelloResponse*. This initializes the connection, readying it for the exchange of further messages.

HelloRequest and *HelloResponse* can be supplemented by a list of supported WWKS functions. Both systems thereby indicate which WWKS functions they can process. If there is no listing of supported WWKS functions, the receiving system assumes that the sending system supports the full set of WWKS 2.0 messages.

Optionally, a tenant identifier can be sent in the *HelloRequest.* This identifier is used to distinguish between individual pharmacy IT systems if an automated storage and retrieval system is addressed by several pharmacy IT system at once. All subsequent *Request* messages are then processed in the context of this tenant.

Sequence



6.1.1 HelloRequest

Structure

```
<WWKS>
  <HelloRequest>
    <Subscriber>
      <Capability/>
    </Subscriber>
  </HelloRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
HelloRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	Message ID. This is returned in *HelloResponse*.

Element	M/O	Data type	Description
Subscriber	M	Tag	Identification of sender (pharmacy IT system)
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	Integer 32-bit >0	ID of the message sender (here the pharmacy IT system). The ID is used in all further messages to identify their senders and recipients. The ID of the pharmacy IT system is defined in the pharmacy IT system. Numbers between 899 and 1000 are reserved for Rowa.
Type	M	String	Type of sending system. Possible value: "IMS" for the pharmacy IT system "Robot" for the automated storage and retrieval system
Manufacturer	M	String	Name of the manufacturer of the sending system
ProductInfo	M	String	Name of the software of the sending system
VersionInfo	M	String	Version of the software of the sending system
TenantId	O	String	Tenant identifier of the pharmacy IT system. Is only used if several pharmacy IT systems are connected with an automated storage and retrieval system.

Element	M/O	Data type	Description
Capability	O	Tag	List of supported WWKS 2 functions. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Name	M	String	Name of the supported WWKS 2 function. Possible value: "KeepAlive" "Status" "Input" "InitiateInput" "ArticleMaster" "StockDelivery" "StockInfo" "Output" "TaskInfo" "TaskCancel" "Configuration" "StockLocationInfo" These designations correspond to the first part of the name of the lead message elements. Example: "Input" means that *InputRequest*, *InputResponse* and *InputMessage* are supported.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <HelloRequest Id="1001">
    <Subscriber Id="100" Type="IMS" Manufacturer="IT-SysProvider"
                ProductInfo="PharmaProg 2013" VersionInfo="1.4.0"
                TenantId="XAB12345">
      <Capability Name="KeepAlive"/>
      <Capability Name="Status"/>
      <Capability Name="Input"/>
      <Capability Name="InitiateInput"/>
      <Capability Name="ArticleMaster"/>
      <Capability Name="StockDelivery"/>
      <Capability Name="StockInfo"/>
      <Capability Name="Output"/>
      <Capability Name="TaskInfo"/>
      <Capability Name="TaskCancel"/>
      <Capability Name="Configuration"/>
      <Capability Name="StockLocationInfo"/>
    </Subscriber>
  </HelloRequest>
</WWKS>
```

Library

TCP/IP connection setup, including initial sending of the *HelloRequest* and receiving of the *HelloResponse* is implemented by the `Connect` method. A mandatory operator is the computer name, or IP address, of the Rowa system. This automatically connects to port 6050. Example:

```
storageSystem.Connect("192.168.64.6");
```

If an alternative port was specified (here 6053), the connection would look like this:

```
storageSystem.Connect("192.168.64.6", 6053);
```

Logging:

It is possible to log all network traffic in so-called WWI files (**.wwi*). These files are useful for troubleshooting or for analysis of customers' problems. A WWI file is a UTF-8-coded text file containing the WWKS 2 messages received and sent. A time stamp is set before each logged message. An *S:* is also prefixed when the message has been sent, or *R:* when it has been received. WWI files can be displayed with any text editor.

In the following example, WWI files with the base name *networktrace* are created in the folder *C:\mylogs*:

```
storageSystem.Connect("192.168.64.6", 6050, "c:\\mylogs", "networktrace");
```

The library creates one new WWI file per day when communication is running. If a file already exists for the current day, any further communications are appended to the end of the file. No logged messages are overwritten or deleted.

6.1.2 HelloResponse

Structure

```
<WWKS>
  <HelloResponse>
    <Subscriber>
      <Capability/>
    </Subscriber>
  </HelloResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
HelloResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID sent in the *HelloRequest*

Element	M/O	Data type	Description
Subscriber	M	Tag	Identifier of the subscriber
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	Integer 32-bit >0	ID of the message sender (here the automated storage and retrieval system). The ID is used in all further messages to identify their senders and recipients.
Type	M	String	Type of sending system. Possible value: "IMS" for the pharmacy IT system "Robot" for the automated storage and retrieval system
Manufacturer	M	String	Name of the manufacturer of the sending system
ProductInfo	M	String	Name of the software of the sending system
VersionInfo	M	String	Version of the software of the sending system

Element	M/O	Data type	Description
Capability	O	Tag	List of supported WWKS 2 functions. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Name	M	String	Name of the supported WWKS 2 function. Possible value: "KeepAlive" "Status" "Input" "InitiateInput" "ArticleMaster" "StockDelivery" "StockInfo" "Output" "TaskInfo" "TaskCancel" "Configuration" "StockLocationInfo" These designations correspond to the first part of the name of the lead message elements. Example: "Input" means that *InputRequest*, *InputResponse* and *InputMessage* are supported.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <HelloResponse Id="1001">
    <Subscriber Id="999" Type="Robot" Manufacturer="Becton Dickinson Rowa Germany
                GmbH" ProductInfo="Mosaic" VersionInfo="2.0.1">
      <Capability Name="KeepAlive"/>
      <Capability Name="Status"/>
      <Capability Name="Input"/>
      <Capability Name="InitiateInput"/>
      <Capability Name="ArticleMaster"/>
      <Capability Name="StockDelivery"/>
      <Capability Name="StockInfo"/>
      <Capability Name="Output"/>
      <Capability Name="TaskInfo"/>
      <Capability Name="TaskCancel"/>
      <Capability Name="Configuration"/>
      <Capability Name="StockLocationInfo"/>
    </Subscriber>
  </HelloResponse>
</WWKS>
```

Library

See *HelloRequest*.

6.2 Keepalive

Lead elements

KeepAliveRequest

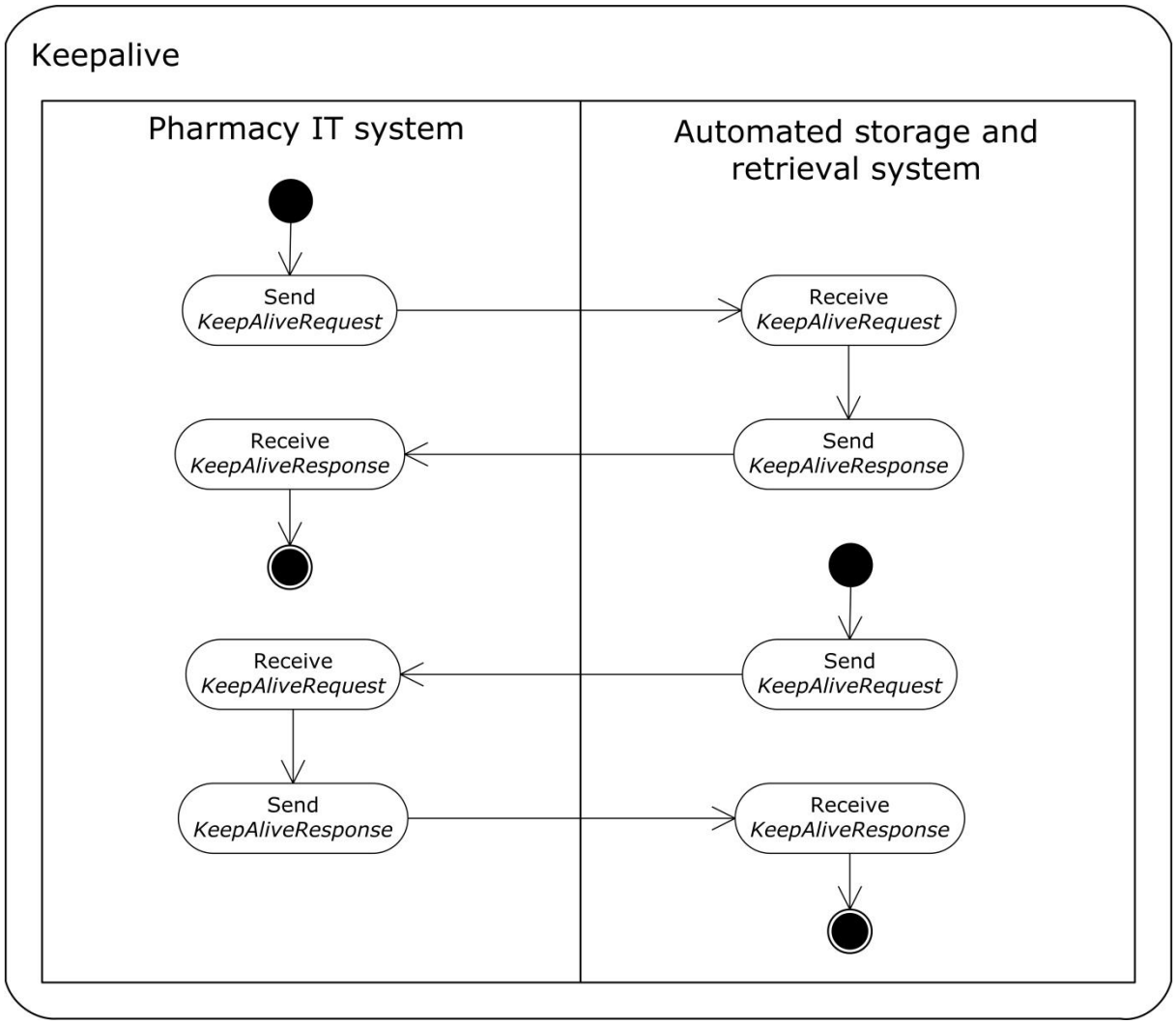
KeepAliveResponse

Usage

A Keepalive request can be sent at any time by both systems to check whether the transport channel underlying the connection is still active. This may be useful under the following conditions:

- When the transport channel is potentially unstable (such as with UMTS, GPRS).
- When the network infrastructure being used comprises many active components (e.g. managed switches, routers) and it may be that one of the components has cut the connection.

Sequence



6.2.1 KeepAliveRequest

Structure

```
<WWKS>
  <KeepAliveRequest/>
</WWKS>
```

Elements

Element	M/O	Data type	Description
KeepAliveRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	Message ID. This is returned in the *KeepAliveResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *KeepAliveRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *KeepAliveRequest*

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <KeepAliveRequest Id="1003" Source="100" Destination="999"/>
</WWKS>
```

Library

The library responds automatically and transparently to every *KeepAliveRequest* from the automated storage and retrieval system.
The library does not currently support a *KeepAliveRequest* from the pharmacy IT system.

6.2.2 KeepAliveResponse

Structure

```
<WWKS>
  <KeepAliveResponse/>
</WWKS>
```

Elements

Element	M/O	Data type	Description
KeepAliveResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	Message ID. This is the same as the one sent in the *KeepAliveResponse* message.
Source	M	Integer 32-bit >0	ID of the system sending the *KeepAliveResponse*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *KeepAliveResponse*

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <KeepAliveResponse Id="1003" Source="999" Destination="100"/>
</WWKS>
```

Library

The library responds automatically and transparently to every *KeepAliveRequest* from the automated storage and retrieval system.
The library does not currently support a *KeepAliveRequest* from the pharmacy IT system.

6.3 System status

Lead elements

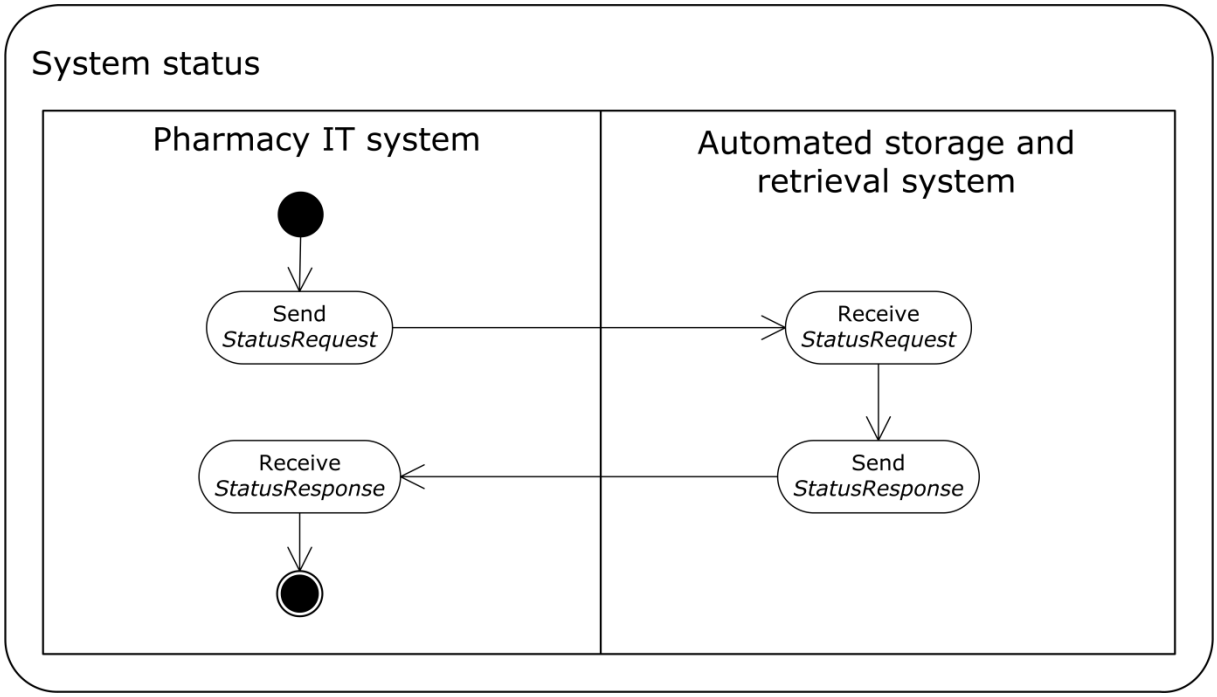
StatusRequest

StatusResponse

Usage

StatusRequest is used to identify the status of the automated storage and retrieval system. In its response, the automated storage and retrieval system indicates whether it is ready for operation and able to execute tasks. This message can be sent as often as desired.

Sequence



6.3.1 StatusRequest

Structure

```
<WWKS>
  <StatusRequest/>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StatusRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	Message ID. This is returned in the *StatusResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *StatusRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StatusRequest*
IncludeDetails	O	Boolean	Optional data, indicates whether the StatusResponse is intended to contain component information. The default value is "False".

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StatusRequest Id="1003" Source="100" Destination="999"/>
</WWKS>
```

Library

The current status of the automated storage and retrieval system can be polled at any time by way of the `State` attribute:

```
StorageSystemState state = storageSystem.State;
```

When polling the status, the library sends a *StatusRequest*, waits for the corresponding *StatusResponse* and evaluates it.

If automatic notification of state changes is required, the user can register for the `StateChanged` event:

```
storageSystem.StateChanged += StorageSystem_StateChanged;
void StorageSystem_StateChanged(IStorageSystem sender, StorageSystemState state)
{
	Console.WriteLine("Storage system changed state to '{0}'.",
				state.ToString());
}
```

6.3.2 StatusResponse

Structure

```
<WWKS>
  <StatusResponse>
      <Component/>
  </StatusResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StatusResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the status request sent in *StatusRequest*
Source	M	Integer 32-bit >0	ID of the system sending the *StatusResponse*
Destination	M	Integer 32-bit >0	Corresponds to the value of Source in the *StatusRequest*.
State	M	String	Designates the current status of the automated storage and retrieval system. Possible values: "Ready" "NotReady"
StateText	O	String	Any text for debugging and logging information

Element	M/O	Data type	Description
Component	O	Tag	Component information follows. This element may occur multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Type	M	String	Type of component. Possible values: "StorageSystem" "BoxSystem"
Description	M	String	Component description. This text can be displayed on program interfaces.
State	M	String	Designates the current status of the component. Possible values: "Ready" "NotReady"

State Text	O	String	Any text for debugging and logging information

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StatusResponse Id="1003" Source="999" Destination="100" State="NotReady">
       <Component Type="StorageSystem" Description="Vmax 1" State="Ready"/>
       <Component Type="StorageSystem" Description="Vmax 2" State="Ready"/>
       <Component Type="BoxSystem" Description="Box System" State="NotReady"
                  StateText="Box system plc is turned off."/>
  </StatusResponse>
</WWKS>
```

Library

See *StatusRequest*.

6.4 Input system

Lead elements

InputRequest

InputResponse

InputMessage

Usage

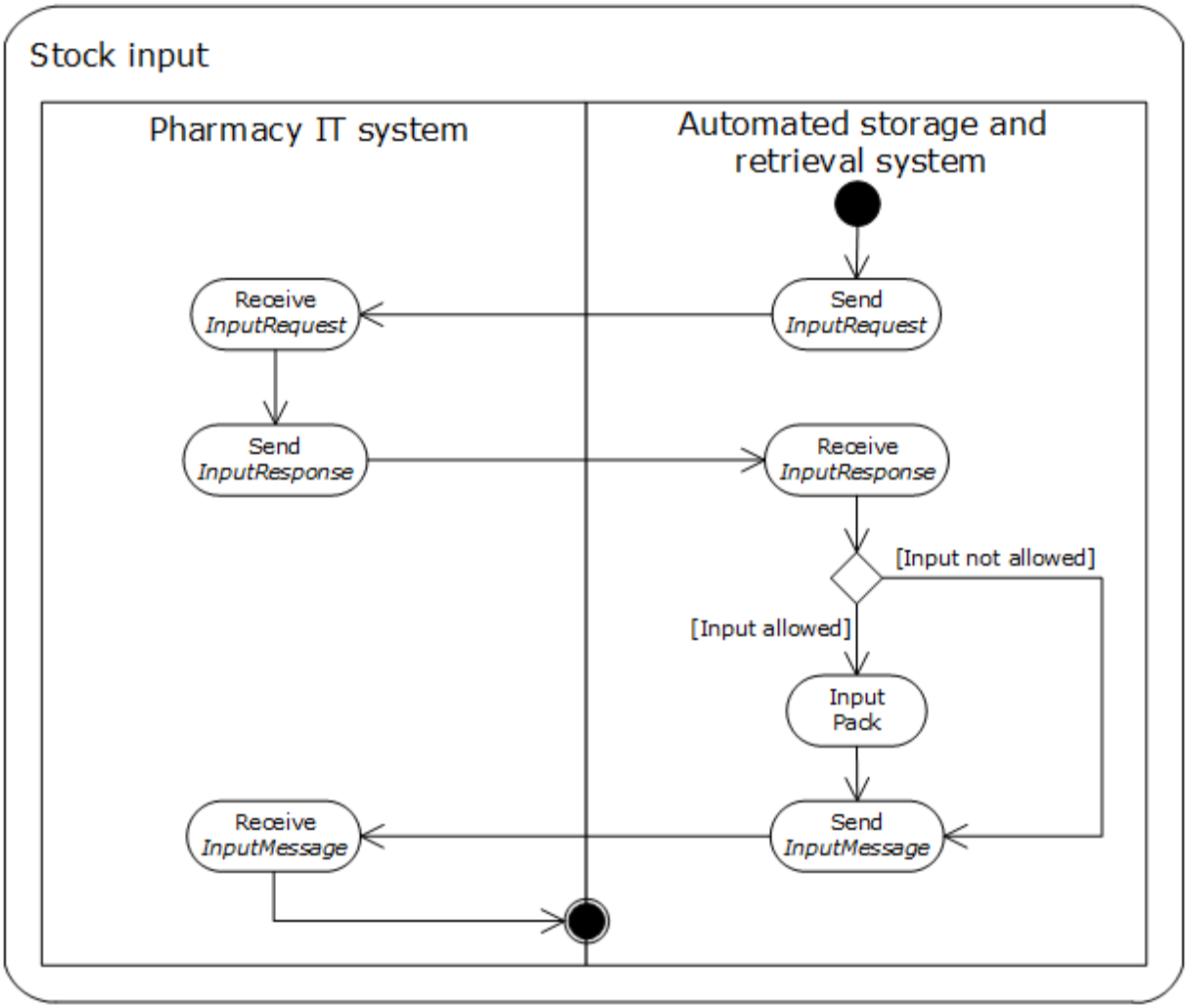
When a user or an automatic stock input system is wanting to store a pack in the automated storage and retrieval system, the *InputRequest* is sent from the automated storage and retrieval system to the pharmacy IT system. The pharmacy IT system must respond to the stock input request with the *InputResponse*. The response from the pharmacy IT system may also send additional article data to the automated storage and retrieval system.

If stock input is allowed, it is executed. The automated storage and retrieval system then sends the *InputMessage* to the pharmacy IT system, informing it of the changed inventory level.

If stock input is not allowed or fails for technical reasons, the automated storage and retrieval system sends the *InputMessage* with the attribute *Input*="Aborted" to the pharmacy IT system.

If the stock input is rejected with the reasons "RejectedNoExpiryDate" or "RejectedNoBatchNumber", the automated storage and retrieval system can resend an *InputRequest* message with the same ID and the missing information.

Sequence


Stock input
Pharmacy IT system
Automated storage and retrieval system
Receive InputRequest
Send InputRequest
Send InputResponse
Receive InputResponse
[Input not allowed]
[Input allowed]
Input Pack
Receive InputMessage
Send InputMessage

6.4.1 InputRequest

Structure

```
<WWKS>
  <InputRequest>
    <Article>
      <Pack/>
    </Article>
  </InputRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
InputRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock input process. This ID is returned in the *InputResponse* and is used in the associated *InputMessage*.
Source	M	Integer 32-bit >0	ID of the system sending the *InputRequest*
Destination	M	Integer 32-bit >0	ID of the system receiving this message
IsNewDelivery	O	Boolean	This flag identifies the stock input request as part of a stock delivery. "True" means that a stock delivery is taking place. The attribute *DeliveryNumber* in the element *Pack* should then contain the stock delivery number. Blank stock delivery numbers (e.g. "") are possible in some customer-specific scenarios. "False" means that the stock input relates to a stock return. The default value is "False".
SetPickingIndicator	O	Boolean	This flag forces definition of the article as capable of automated handling after the pharmacy IT system has initially rejected stock input with the reason "RejectedNoPickingIndicator". This case occurs primarily on initial stocking of the automated storage and retrieval system. The default value is "False".

Element	M/O	Data type	Description
Article	M	Tag	Article information follows.

Element	M/O	Data type	Description
Pack	M	Tag	Pack information follows. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Index	O	Integer 32-bit >=0	If the stock input request comprises multiple packs, a pack index is sent here.
ScanCode	M	String	Barcode of the pack being placed into stock. This code is checked by the pharmacy IT system and compared against internal data. Based on the information contained in the code, the pharmacy IT system can decide whether the pack can be placed into stock.
DeliveryNumber	O	String	Stock delivery number for this pack. This data is only required if the attribute *IsNewDelivery* of the element *InputRequest* has the value "True".
BatchNumber	O	String	Batch number of the pack. This attribute is used when the picker has entered a batch number.
ExternalId	O	String	External ID. Additional identifying attribute. This can be used, for example, to store additional information about a pack (e.g. the serial number).
ExpiryDate	O	String	Expiry date of the pack in format YYYY-MM-DD. This attribute is used when the picker has entered an expiry date.
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened). This attribute is used for the stock input of opened packs.
StockLocationId	O	String	ID of the selected stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the requesting machine intended to be used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputRequest Id="1002" Source="999" Destination="100" IsNewDelivery="True">
    <Article>
      <Pack Index="0" DeliveryNumber="463526"
       ScanCode="0004-56-034-G|00007|T|Omepra0004|PalH09051200001|051112|XXXXXX"/>
    </Article>
  </InputRequest>
</WWKS>
```

Library

To handle a stock input process, the user must register for the `PackInputRequested` and `PackStored` events:

```
storageSystem.PackInputRequested += StorageSystem_PackInputRequested;
storageSystem.PackStored += StorageSystem_PackStored;
```

When one or more packs is awaiting stock input, the automated storage and retrieval system sends an *InputRequest*. This triggers the `PackInputRequested` event. As part of the event handling method, the pharmacy IT system must decide whether the requested packs may be placed into stock or not. At the same time, additional article information about the packs being placed into stock must be delivered. The pack objects being placed into stock contain as a minimum the pack's scan code, and optionally also other attributes such as an expiry date or batch number.

```
void StorageSystem_PackInputRequested(IStorageSystem sender, IInputRequest request)
{
	foreach (var pack in request.Packs)
	{
	// Check the scan code of the pack and generate an appropriate article
	// identifier. In this sample we use the pack scan code as article identifier
	// for simplicity reasons,
	string articleId = pack.ScanCode;

	// PUT DECISION LOGIC HERE

	// define article information for this pack
	pack.SetArticleInformation(articleId,
	                           "My Article Name",
	                           "Dosage Form",
	                           "Packaging Unit");
	// allow pack input
	pack.SetHandling(InputHandling.Allowed);
	}
	// send InputResponse
	request.Finish();

}
```

A pack can be rejected by the call `pack.SetHandling(InputHandling.Rejected)`. Input to a refrigerated storage location can be forced by the call `pack.SetHandling(InputHandling.AllowedForFridge)`. There are more handling definitions which, for example, force specification of a batch number or an expiry date for the scanned pack. The code add-on of the development environment shows additional defined values of *`InputHandling`*. The naming of values may also be derived from the descriptions of the XML messages (see following tables).

6.4.2 InputResponse

Structure

```
<WWKS>
  <InputResponse>
    <Article>
      <Pack>
        <Handling/>
      </Pack>
    </Article>
  </InputResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
InputResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock input process. This ID was sent in the *InputRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *InputResponse*
Destination	M	Integer 32-bit >0	Corresponds to the value of Source in the associated *InputRequest*.
IsNewDelivery	O	Boolean	This flag identifies the stock input request as part of a stock delivery. The same value as in the associated *InputRequest* is used here.

Element	M/O	Data type	Description
Article	M	Tag	Article information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	O	String	The article ID to be used on stock input into the automated storage and retrieval system. In the code examples below the ID is composed from data taken from the transmitted barcode. It is up to the pharmacy IT system how the article ID is composed.
Name	O	String	Name of article. The automated storage and retrieval system displays the name in its own inventory view (GUI).

DosageForm	O	String	Dosage form of the article. The automated storage and retrieval system displays the name in its own inventory view (GUI).
PackagingUnit	O	String	Packaging unit of the article. The automated storage and retrieval system displays the name in its own inventory view (GUI).
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which are contained in a full pack of the article. The value "0" means that the number of units is unknown.

Element	**M/O**	**Data type**	**Description**
Pack	M	Tag	Pack information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Index	O	Integer 32-bit >=0	Index number of the pack. Used only if the *InputResponse* covers multiple packs. The index number corresponds to the one in the *InputRequest*.
DeliveryNumber	O	String	Stock delivery number specified in the *InputRequest*
BatchNumber	O	String	Batch number to be saved for this pack
ExternalId	O	String	External ID. Additional identifying attribute. This can be used, for example, to store additional information about a pack (e.g. the serial number).
ExpiryDate	O	String	Expiry date in format YYYY-MM-DD
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The pharmacy IT system can overwrite the original value from the *InputRequest* with this. The value "0" means that the pack is full (not opened).
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system to be used. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.

Element	**M/O**	**Data type**	**Description**

Handling	M	Tag	Instructions for handling of the pack follow. This element can occur once per pack.
Attributes	**M/O**	**Data type**	**Description and Values**
Input	M	String	Handling instructions for the pack. Possible values: "Allowed" if stock input is allowed "AllowedForFridge" if the article needs to be refrigerated "Rejected" if stock input is not allowed "RejectedNoExpiryDate" if stock input is not allowed because the expiry date was not specified "RejectedNoPickingIndicator" if stock input is not allowed because the article is not declared as capable of automated handling "RejectedNoBatchNumber" if stock input is not allowed because the batch number was not specified in the automated storage and retrieval system "RejectedNoStockLocation" if stock input is not allowed because the stock location ID was not specified in the automated storage and retrieval system "RejectedInvalidStockLocation" if stock input is not allowed because the stock location ID specified in the automated storage and retrieval system is not authorized for this article
Text	O	String	Any text detailing additional instructions for handling of the pack. The text is normally used to detail the reasons for rejection of packs. The text might be displayed on the automated storage and retrieval system, and so should be capable of being localized.

Example of allowing stock input of a pack

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputResponse Id="1002" Source="100" Destination="999">
    <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
             DosageForm="LOE" PackagingUnit="1X2.5 ML">
      <Pack Index="0" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2012-11-05">
        <Handling Input="Allowed" Text="Pack input allowed."/>
      </Pack>
    </Article>
  </InputResponse>
</WWKS>
```

Example of refusing stock input of a pack

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputResponse Id="1002" Source="100" Destination="999">
    <Article>
      <Pack Index="0">
        <Handling Input="Rejected" Text="Pack input forbidden."/>
      </Pack>
    </Article>
  </InputResponse>
</WWKS>
```

Library

See *InputRequest*.

6.4.3 InputMessage

Structure

```
<WWKS>
  <InputMessage>
    <Article>
      <Pack>
        <Handling/>
      </Pack>
    </Article>
  </InputMessage>
</WWKS>
```

Elements

Element	M/O	Data type	Description
InputMessage	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock input process. This ID was sent in the *InputRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *InputMessage*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *InputMessage*
IsNewDelivery	O	Boolean	This flag identifies the stock input as part of a stock delivery. The same value as in the associated *InputRequest* is used here.

Element	M/O	Data type	Description
Article	M	Tag	Article information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	The article ID sent in the *InputResponse*
Name	O	String	Name of the article sent in the *InputResponse*
DosageForm	O	String	The dosage form sent in the *InputResponse*
PackagingUnit	O	String	The packaging unit sent in the *InputResponse*
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which are contained in a full pack of the article. The value "0" means that the

			number of units is unknown. The value corresponds to the one sent in the *InputResponse*.

Element	**M/O**	**Data type**	**Description**
Pack	M	Tag	Pack information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Index	O	Integer 32-bit >=0	Index number of the pack. Used only if the *InputMessage* covers multiple packs. The index number corresponds to the one in the *InputRequest*.
Id	M	Integer 64-bit >=0	Internal pack ID. If the pack has not been placed into stock, the value is "0".
DeliveryNumber	O	String	Stock delivery number specified in the *InputRequest*
BatchNumber	O	String	Batch number saved for this pack
ExternalId	O	String	External ID specified in the *InputResponse* and saved for this pack
ExpiryDate	O	String	Expiry date in format YYYY-MM-DD
StockInDate	O	String	Input date in the format YYYY-MM-DD
ScanCode	O	String	Barcode of the input pack
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened).
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack. The default value is "Cuboid". "Cylinder" is also supported.
State	O	String	Status of the pack. This data is required when multiple automatic storage machines are connected. The default value "Available" means that the pack is currently available for output. "NotAvailable" is also supported.

StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Element	**M/O**	**Data type**	**Description**
Handling	M	Tag	Input result follows. This element can occur once per pack.
Attributes	**M/O**	**Data type**	**Description and Values**
Input	M	String	Outcome of the stock input. Possible values: "Completed" if the pack was placed into stock "Aborted" if an error occurred or the stock input was aborted
Text	O	String	Any text for debugging and logging information

Example of successful stock input

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputMessage Id="1002" Source="999" Destination="100">
    <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
             DosageForm="LOE" PackagingUnit="1X2.5 ML">
      <Pack Index="0" Id="565362" BatchNumber="Omepra0004"
            ExternalId="PalH09051200001" ExpiryDate="2012-11-05"
            Depth="50" Width="50" Height="50" Shape="Cuboid"
            State="Available">
        <Handling Input="Completed" Text="Pack input completed."/>
      </Pack>
    </Article>
  </InputMessage>
</WWKS>
```

Example of stock input aborted because of rejection by pharmacy IT system

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputMessage Id="1002" Source="999" Destination="100">
    <Article>
      <Pack Index="0" Id="0">
        <Handling Input="Aborted" Text="Pack input aborted."/>
      </Pack>
    </Article>
  </InputMessage>
</WWKS>
```

Example of stock input aborted by user

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InputMessage Id="1002" Source="999" Destination="100">
    <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
             DosageForm="LOE" PackagingUnit="1X2.5 ML">
      <Pack Index="0" Id="0" BatchNumber="Omepra0004"
            ExternalId="PalH09051200001" ExpiryDate="2012-11-05">
        <Handling Input="Aborted" Text="Pack input aborted."/>
      </Pack>
    </Article>
  </InputMessage>
</WWKS>
```

Library

If packs have been successfully placed into stock, the *InputMessage* sent by the automated storage and retrieval system sends triggers the `PackStored` event. In the event handling method the pharmacy IT system can then update its inventory information. If the stock input process was aborted, no event is triggered because the inventory was not changed.

```
void StorageSystem_PackStored(IStorageSystem sender, IArticle[] articleList)
{
	foreach (var article in articleList)
	{
		foreach (var pack in article.Packs)
		{
		Console.WriteLine("Pack '{0}' has been stored in the storage system.",
		pack.Id);
		}
	}
}
```

6.5 Stock input initiation

Lead elements

InitiateInputRequest

InitiateInputResponse

InitiateInputMessage

Usage

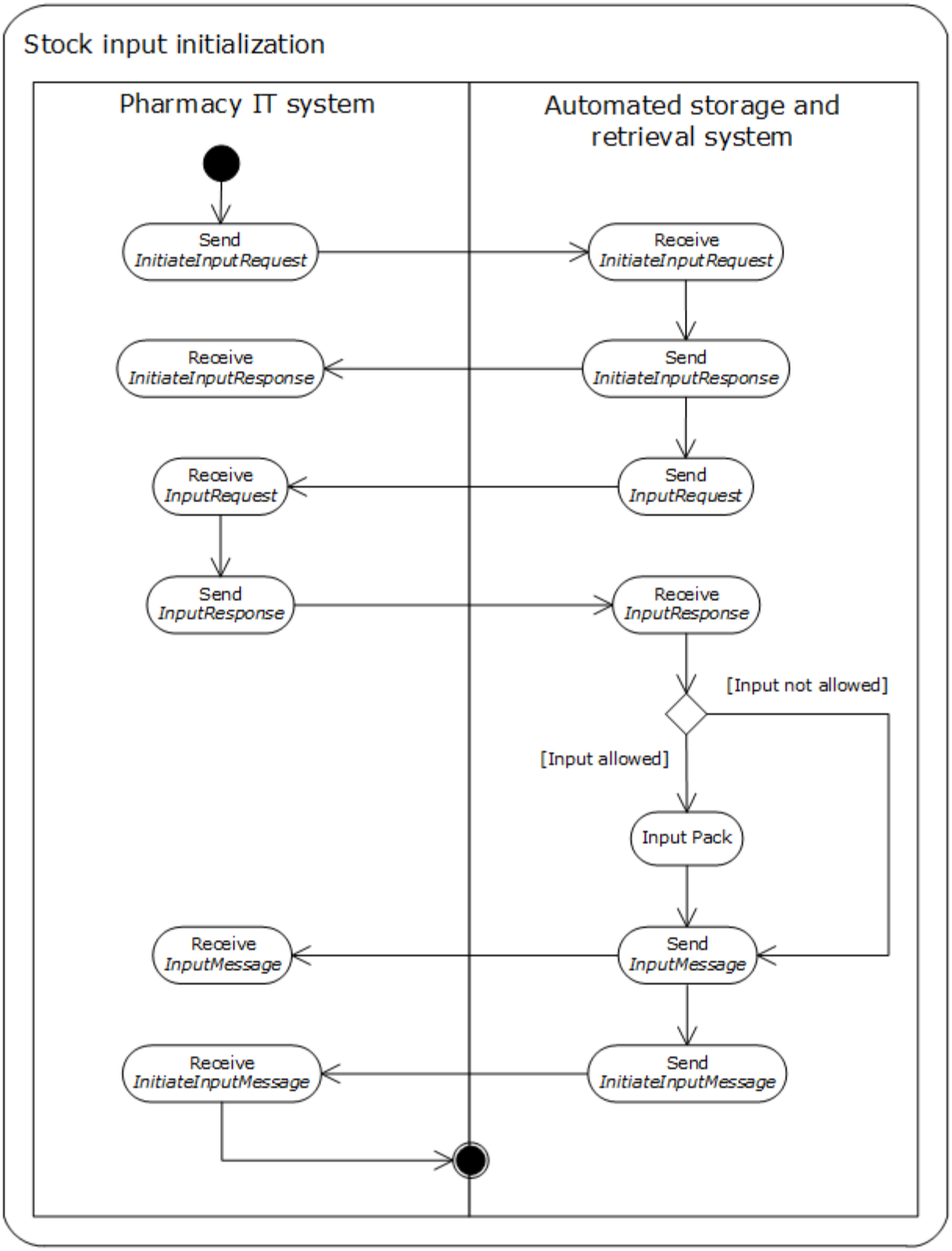
The pharmacy IT system can explicitly trigger an input process by sending an *InitiateInputRequest.* This message and the appertaining messages *InitiateInputResponse* and *InitiateInputMessage* are used when the automated storage and retrieval system is connected to a third party automated input system or a specific input behaviour is requested for other reasons.

It is the scope of pharmacy IT system the to make sure that the packs have been positioned correctly at a defined transfer point to the automated storage and retrieval system and are thus ready to be input by the automated storage and retrieval system. This can be, for example, realized through instructions in the user interface of the pharmacy IT system.

The triggered input process corresponds to the regular input process as described in the section *Stock input*. The entire process is completed as soon as the triggered input process is completed.

The automated storage and retrieval system can either return the *InitiateInputResponse* immediately or wait for the response to the corresponding input request of the pharmacy IT system. The process demonstrated in the flowchart corresponds to the first case.

Sequence


Stock input initialization
Pharmacy IT system
Automated storage and retrieval system
Send InitiateInputRequest
Receive InitiateInputRequest
Receive InitiateInputResponse
Send InitiateInputResponse
Receive InputRequest
Send InputRequest
Send InputResponse
Receive InputResponse
[Input not allowed]
[Input allowed]
Input Pack
Receive InputMessage
Send InputMessage
Receive InitiateInputMessage
Send InitiateInputMessage

6.5.1 InitiateInputRequest

Structure

```
<WWKS>
  <InitiateInputRequest>
    <Details/>
    <Article>
      <Pack/>
    </Article>
  </InitiateInputRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
InitiateInputRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock input initiation process. This ID is returned in the *InitiateInputResponse* and used in the corresponding *InitiateInputMessage*.
Source	M	Integer 32-bit >0	ID of the system sending *InitiateInputRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to start the stock input. Example: 999 = Rowa Vmax 1 998 = Rowa Vmax 2 The values between 899 and 1000 are reserved for Rowa.
IsNewDelivery	O	Boolean	This flag specifies whether the triggered stock input is part of a new delivery. "True" means that a stock delivery is taking place. The attribute *DeliveryNumber* in the element *Pack* should then contain the stock delivery number. Blank stock delivery numbers (e.g. "") are possible in some customer-specific scenarios. "False" means that the triggered stock input relates to a stock return. The default value is "False".
SetPickingIndicator	O	Boolean	This flag forces definition of the article as capable of automated handling after the pharmacy IT system has initially rejected stock input with the reason "RejectedNoPickingIndicator". This case occurs primarily on initial stocking of the automated storage and retrieval system.

			The default value is "False".

Element	**M/O**	**Data type**	**Description**
Details	M	Tag	Stock input information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
InputSource	M	Integer 32-bit	Identifies the transfer point to the automated storage and retrieval system intended to be used for the stock input.
InputPoint	O	Integer 32-bit	Detailed information on the transfer point used (e.g. belt number)

Element	**M/O**	**Data type**	**Description**
Article	M	Tag	Article information follows.

Element	**M/O**	**Data type**	**Description**
Pack	M	Tag	Pack information follows. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Index	O	Integer 32-bit >=0	If the stock input request comprises multiple packs of an article, a pack index is sent here.
ScanCode	M	String	Barcode of the pack being placed into stock. This code is checked by the pharmacy IT system and compared against internal data. Based on the information contained in the code, the pharmacy IT system can decide whether the pack can be placed into stock.
DeliveryNumber	O	String	Stock delivery number for this pack. This information is only necessary when the attribute *IsNewDelivery* of the element *InitiateInputRequest* has the value "True".
BatchNumber	O	String	Batch number of the pack. This attribute is used when a batch number is to be sent along with the stock input request.
ExternalId	O	String	External ID. Additional identifying attribute. This attribute is used when an external ID is to be sent along with the input request.

ExpiryDate	O	String	Expiry date of the pack in format YYYY-MM-DD. This attribute is used when an expiry date is to be sent along with the stock input request.
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened). This attribute is used for the stock input of opened packs.
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack. The default value is "Cuboid". "Cylinder" is also supported.

StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system that is intended to be used for storing the pack. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the requesting machine intended to be used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InitiateInputRequest Id="1003" Source="100" Destination="999">
    <Details InputSource="3" InputPoint="1"/>
    <Article>
      <Pack Index="0" ScanCode="0004-56-034-G|00007|T|Omepra0004|PalH09051200001"
            Depth="50" Width="50" Height="50" Shape="Cuboid"/>
    </Article>
  </InitiateInputRequest>
</WWKS>
```

Library

To trigger a stock input process, a new stock input process object is created by using the `CreateInitiateInputRequest` method. Then the `AddInputPack` method is used to define the pack to be input abd the `Start` method is used to start the stock input process. By registering for the `Finished` event, it is possible to react to the completion of the process.

```
// create new input process with id 111, input source 3
// and input point 1 for destination 998
var initiateInput = storageSystem.CreateInitiateInputRequest(111, 3, 1, 998);

// define a pack with the scan code "584638439" for input
initiateInput.AddInputPack("584638439");

// register for "Finished" event.
initiateInput.Finished += OnInitiateInput_Finished;

// start the input process
initiateInput.Start();
```

Within the scope of the event handling method for the `Finished` event, it is possible to determine details for the completed stock input process. As it is possible to define several packs per stock input, errors can occur for each pack. For this reason, the error information in the response are pack-based.

```
void OnInitiateInput_Finished(object sender, EventArgs e)
{
    var initiateInput = sender as IInitiateInputRequest;

    if (initiateInput.State == InitiateInputRequestState.Completed)
    {
        // everything is ok -> access the list of processed articles
        // with detailed article and pack information of the stored packs
        var inputArticles = initiateInput.InputArticles;
    }
    else
    {
        // at least one pack failed to input
        for each (var article in initiateInput.InputArticles)
        {
            for each (var pack in article.Packs)
            {
                string errorText;
                InputErrorType errorType;

                if (initiateInput.GetProcessedPackError(pack,
                                                        out errorType,
                                                        out errorText))
                {
                    // process pack error details
                }
            }
        }
    }
}
```

The `InputArticles` property can already be set when the `Start` method is completed – depending on the behavior and the configuration of the connect automated storage and retrieval system – and contains the article information of the pack to be input. This can be used to display status information in the user interface of the pharmacy IT system.

6.5.2 InitiateInputResponse

Structure

```
<WWKS>
  <InitiateInputResponse>
    <Details/>
    <Article>
      <Pack/>
    </Article>
  </InitiateInputResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
InitiateInputResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock input initiation process
Source	M	Integer 32-bit >0	ID of the system sending *InitiateInputResponse*
Destination	M	Integer 32-bit >0	ID of the system that sent the original *InitiateInputRequest.*
IsNewDelivery	O	Boolean	This flag specifies whether the triggered stock input is part of a new delivery. "True" means that a stock delivery is taking place. The attribute *DeliveryNumber* in the element *Pack* should then contain the stock delivery number. Blank stock delivery numbers (e.g. "") are possible in some customer-specific scenarios. "False" means that the triggered stock input relates to a stock return. The default value is "False". Here the same value is to be seen as in the associated *InitiateInputRequest*.
SetPickingIndicator	O	Boolean	This flag forces definition of the article as capable of automated handling after the pharmacy IT system has initially rejected stock input with the reason "RejectedNoPickingIndicator". This case occurs primarily on initial stocking of the automated storage and retrieval system. The default value is "False". Here the same value is to be seen as in the associated *InitiateInputRequest*.

Element	M/O	Data type	Description
Details	M	Tag	Stock input information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
InputSource	M	Integer 32-bit	Identifies the transfer point to the automated storage and retrieval system intended to be used for the stock input.
InputPoint	O	Integer 32-bit	Detailed information on the transfer point used (e.g. belt number). If this property was not defined in the associated *InitiateInputRequest*, the automated storage and retrieval system can here return the detailed transfer point that was automatically determined.
Status	M	String	Status of the triggered stock input process. Within the *InitiateInputResponse,* only the values "Accepted" and "Rejected" are permitted. If the process was canceled with the "Rejected" status, the process is completed and no further messages are sent.

Element	M/O	Data type	Description
Article	M	Tag	Article information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	O	String	Article ID of the pack that was initiated for stock input.
Name	O	String	Name of the article for the pack that was initiated for stock input.
DosageForm	O	String	Dosage from of the article for the pack that was initiated for stock input.
PackagingUnit	O	String	Packaging unit from of the article for the pack that was initiated for stock input.
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which are contained in a full pack of the article. The value "0" means that the number of units is unknown.

Element	M/O	Data type	Description
Pack	M	Tag	Pack information follows. This element can be used multiple times.

Attributes	M/O	Data type	Description and Values
Index	O	Integer 32-bit >=0	If the stock input initiation request comprises multiple packs of an article, a pack index is sent here. Here the same value is to be seen as in the associated *InitiateInputRequest*.
ScanCode	M	String	Barcode of the pack being placed into stock. This code is checked by the pharmacy IT system and compared against internal data. Based on the information contained in the code, the pharmacy IT system can decide whether the pack can be placed into stock. Here the same value is to be seen as in the associated *InitiateInputRequest*.
DeliveryNumber	O	String	Stock delivery number for this pack. Here the same value is to be seen as in the associated *InitiateInputRequest*.
BatchNumber	O	String	Batch number of the pack. Here the same value is to be seen as in the associated *InitiateInputRequest*.
ExternalId	O	String	External ID. Additional identifying attribute. Here the same value is to be seen as in the associated *InitiateInputRequest*.
ExpiryDate	O	String	Expiry date of the pack in format YYYY-MM-DD. Here the same value is to be seen as in the associated *InitiateInputRequest*.

SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened). This attribute is used for the stock input of opened packs. Here the same value is to be seen as in the associated *InitiateInputRequest*.
Depth	O	Integer 32-bit >=0	Depth of the pack in mm. Here the same value is to be seen as in the associated *InitiateInputRequest*. Alternatively, the automated storage and retrieval system can return pack dimensions that it has determined itself (e.g. if there are no dimensions specified in the *InitiateInputRequest*).
Width	O	Integer 32-bit >=0	Width of the pack in mm. Here the same value is to be seen as in the associated *InitiateInputRequest*. Alternatively, the automated storage and retrieval system can return pack dimensions that it has determined itself (e.g. if there are no dimensions specified in the *InitiateInputRequest*).
Height	O	Integer 32-bit >=0	Height of the pack in mm. Here the same value is to be seen as in the associated *InitiateInputRequest*. Alternatively, the automated storage and retrieval system can return pack dimensions that it has determined itself (e.g. if there are no dimensions specified in the *InitiateInputRequest*).
Shape	O	String	Form factor of the pack. The default value is "Cuboid". "Cylinder" is also supported. Here the same value is to be seen as in the associated *InitiateInputRequest*.
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system that is intended to be used for storing the pack. Is only used when an automated storage and retrieval system is divided into several virtual stock locations. Here the same value is to be seen as in the associated *InitiateInputRequest*.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InitiateInputResponse Id="1003" Source="999" Destination="100">
    <Details InputSource="3" InputPoint="1" Status="Accepted"/>
    <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
             DosageForm="LOE" PackagingUnit="1X2.5 ML">
      <Pack Index="0" ScanCode="0004-56-034-G|00007|T|Omepra0004|PalH09051200001"
            Depth="50" Width="50" Height="50" Shape="Cuboid"/>
    </Article>
  </InitiateInputResponse>
</WWKS>
```

Library

See *InitiateInputRequest*.

6.5.3 InitiateInputMessage

Structure

```
<WWKS>
  <InitiateInputMessage>
    <Details/>
    <Article>
      <Pack>
        <Error/>
      </Pack>
    </Article>
  </InitiateInputMessage>
</WWKS>
```

Elements

Element	M/O	Data type	Description
InitiateInputMessage	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock input initiation process. This ID was sent in the *InitiateInputRequest*.
Source	M	Integer 32-bit >0	ID of the system sending *InitiateInputMessage*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *InitiateInputMessage*

Element	M/O	Data type	Description
Details	M	Tag	Stock input information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
InputSource	M	Integer 32-bit >=0	Identifies the transfer point to the automated storage and retrieval system used for the stock input.
InputPoint	O	Integer 32-bit	Detailed information on the transfer point used (e.g. belt number). If this property was not defined in the associated *InitiateInputRequest*, the automated storage and retrieval system can here return the detailed transfer point that was automatically determined.
Status	M	String	Final status of the triggered stock input process. Within the *InitiateInputMessage,* only the values " Completed " and " Incomplete " are permitted. If the process was completed with the "Incomplete" status, at least one of the defined packs could not be input.

Element	M/O	Data type	Description
Article	M	Tag	Article information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	Article ID of the pack that was initiated for stock input
Name	O	String	Name of the article for the pack that was initiated for stock input
DosageForm	O	String	Dosage from of the article for the pack that was initiated for stock input
PackagingUnit	O	String	Packaging unit of the article for the pack that was initiated for stock input
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which are contained in a full pack of the article. The value "0" means that the number of units is unknown.

Element	M/O	Data type	Description
Pack	M	Tag	Pack information follows. This element may occur multiply.

Attributes	M/O	Data type	Description and Values
Index	O	Integer 32-bit >=0	Index number of the pack. Used only if the *InitiateInputMessage* covers multiple packs. The index number corresponds to the one in the *InitiateInputRequest*.
Id	M	Integer 64-bit >=0	Internal pack ID in the automated storage and retrieval system. If the pack has not been placed into stock, the value is "0".
DeliveryNumber	O	String	Stock delivery number specified in the *InitiateInputRequest*
BatchNumber	O	String	Batch number saved for this pack
ExternalId	O	String	External ID specified in the *InitiateInputRequest*
ExpiryDate	O	String	Expiry date in format YYYY-MM-DD
StockInDate	O	String	Input date in the format YYYY-MM-DD
ScanCode	O	String	Barcode of the input pack
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened).
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack. The default value is "Cuboid". "Cylinder" is also supported.
State	O	String	Status of the pack. This data is required when multiple automatic storage machines are connected. The default value "Available" means that the pack is currently available for output. "NotAvailable" is also supported.
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Element	M/O	Data type	Description
Error	O	Tag	Detailed error information follows. This element can occur once per pack.
Attributes	**M/O**	**Data type**	**Description and Values**
Type	M	String	Detailed error that occurred during stock input. Possible values: "Rejected", "RejectedNoExpiryDate", "RejectedInvalidExpiryDate", "RejectedNoPickingIndicator", "RejectedNoBatchNumber", "RejectedNoStockLocation", "RejectedInvalidStockLocation", "QueueFull", "FridgeMissing", "UnknownPackDimensions", "MeasurementError", "PackAcknowledged", "InputBroken" "NoSpaceInMachine", "NoPackDetected"
Text	O	String	Any text for detailed error information

Example of successful stock input

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InitiateInputMessage  Id="1002" Source="999" Destination="100">
    <Details InputSource="3" InputPoint="1" Status="Completed"/>
    <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
             DosageForm="LOE" PackagingUnit="1X2.5 ML">
      <Pack Index="0" Id="565362" BatchNumber="Omepra0004"
            ExternalId="PalH09051200001" ExpiryDate="2012-11-05"
            Depth="50" Width="50" Height="50" Shape="Cuboid"
            State="Available"/>
    </Article>
  </InitiateInputMessage>
</WWKS>
```

Example for an aborted stock input

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <InitiateInputMessage Id="1003" Source="999" Destination="100">
    <Details InputSource="3" InputPoint="1" Status="Incomplete"/>
    <Article>
      <Pack Index="0">
        <Error Type="Rejected" Text="Pack input forbidden."/>
      </Pack>
    </Article>
  </InitiateInputMessage>
</WWKS>
```

Library

See *InitiateInputRequest*.

6.6 Master data

Lead elements

ArticleMasterSetRequest

ArticleMasterSetResponse

StockDeliverySetRequest

StockDeliverySetResponse

Usage

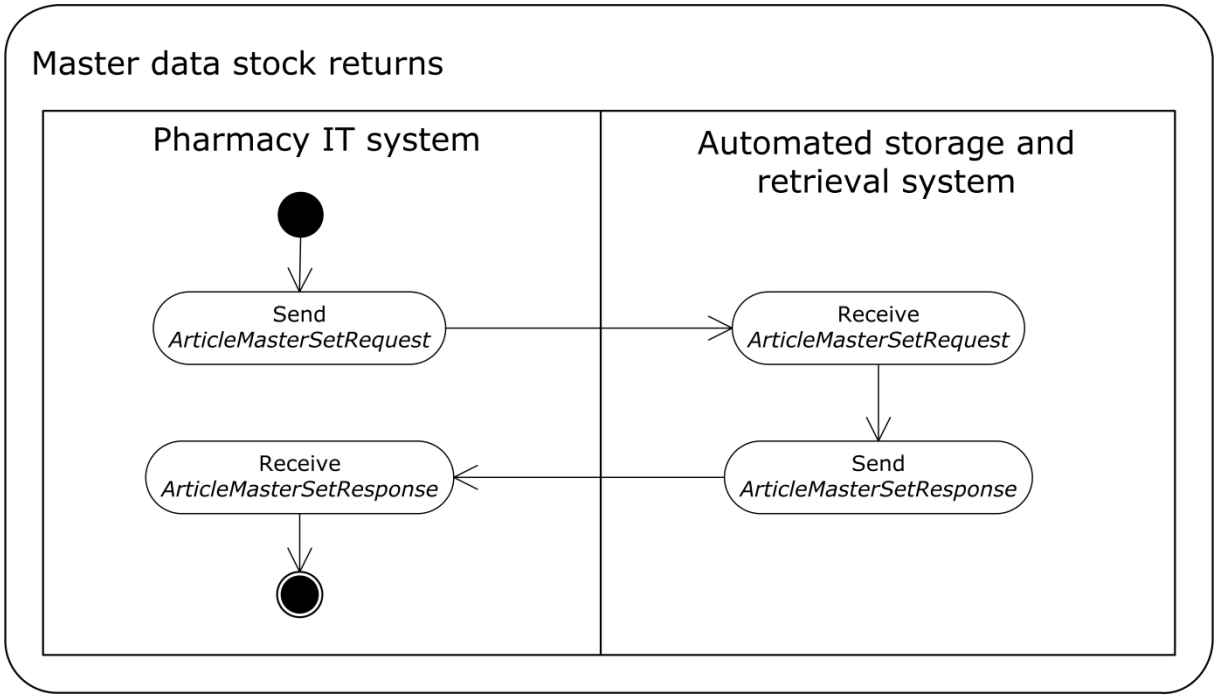
Technical restrictions (such as a slow data connection between the pharmacy IT system and automated storage and retrieval system or long response times) might mean that the pharmacy IT system cannot monitor the stock input process in real time. In such cases, the pharmacy IT can send the automated storage and retrieval system all the data relating to articles capable of being placed into stock in advance. The automated storage and retrieval system then executes the stock input without referring back to the pharmacy IT system.

The data needed for this is:

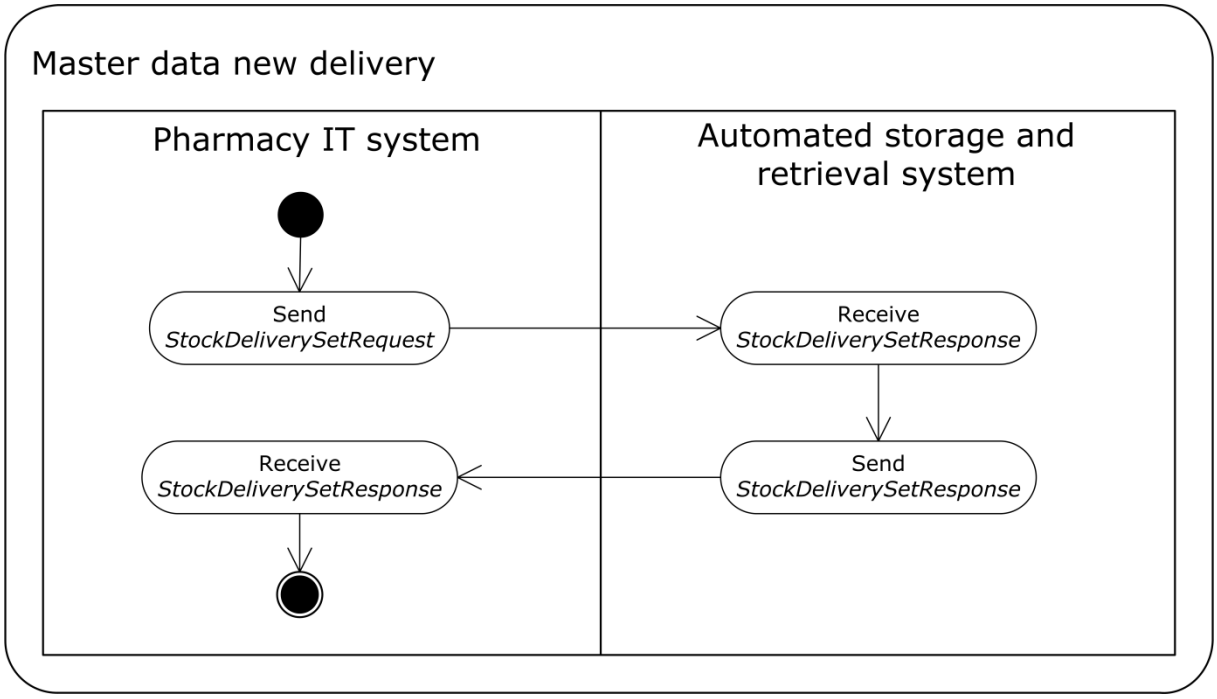
- **Article master**
  The *ArticleMasterSetRequest* contains the descriptions of all articles which can be placed into stock without a stock delivery number. This is applicable primarily for stock returns which might possibly already have been stored in the system.
  An article master already existing in the automated storage and retrieval system is completely overwritten. It is also possible to send the *ArticleMasterSetRequest* with a blank article list, so as to reset the automated storage and retrieval system's article master.
- **Stock delivery**
  The *StockDeliverySetRequest* contains all the data relating to one or more stock deliveries – that is to say, descriptions of all the articles which may be placed into stock under a specific stock delivery number.

The stock deliveries predefined in the automated storage and retrieval system are not overwritten; the newly defined stock deliveries are added to them.

Sequence for article master data



Sequence for stock delivery



6.6.1 ArticleMasterSetRequest

Structure

```
<WWKS>
  <ArticleMasterSetRequest>
    <Article/>
  </ArticleMasterSetRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
ArticleMasterSetRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	O	String	ID of the master data process. This ID is returned in the *ArticleMasterSetResponse*.
Source	O	Integer 32-bit >0	ID of the system sending the *ArticleMasterSetRequest*
Destination	O	Integer 32-bit >0	ID of the system intended to receive the *ArticleMasterSetRequest*

Element	M/O	Data type	Description
Article	O	Tag	Article information follows. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the article. This may be the original barcode of the article or a derivative of it. The pharmacy IT system specifies how the article ID is composed. If the ID is a custom composition, the automated storage and retrieval system must be able to autonomously break the pack barcodes down into the relevant form.
Name	O	String	Name of the article
DosageForm	O	String	Dosage form of the article
PackingUnit	O	String	Packaging unit of the article
RequiresFridge	O	Boolean	Flag indicating whether the article has to be stored refrigerated ("True"). The default value is "False".

MaxSubItemQuantity	O	Integer 32-bit >0	Maximum number of units (e.g. pills or ampules) which might be contained in a full pack of the article. The value "0" means that the number of units is unknown.
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system that has to be used for packs of this article. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine that has to store packs of this article. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <ArticleMasterSetRequest Id="1003" Source="100" Destination="999">
    <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
             DosageForm="LOE" PackagingUnit="1X2.5 ML" RequiresFridge="False"/>
    <Article Id="56473627" Name="PREDNISOLO" DosageForm="TAB" PackagingUnit="20"/>
  </ArticleMasterSetRequest>
</WWKS>
```

Library

If a pharmacy IT system is not permanently connected to the automated storage and retrieval system, or does not respond directly to stock input requests for other reasons, the pharmacy IT system can define an article master with which the automated storage and retrieval system then works autonomously.

The `CreateMasterArticle` method is used to define new master articles. These are collated in a list and then transferred to the automated storage and retrieval system by the `UpdateMasterArticles` method:

```
var masterArticleList = new List<IMasterArticle>();

masterArticleList.Add(storageSystem.CreateMasterArticle("47463736",
                                                        "Article Number 1",
                                                        "Dosage Form",
                                                        "Packaging Unit"));

masterArticleList.Add(storageSystem.CreateMasterArticle("78695739",
                                                        "Article Number 2",
                                                        "Dosage Form",
                                                        "Packaging Unit"));

storageSystem.UpdateMasterArticles(masterArticleList);
```

6.6.2 ArticleMasterSetResponse

Structure

```
<WWKS>
  <ArticleMasterSetResponse>
    <SetResult/>
  </ArticleMasterSetResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
ArticleMasterSetResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the master data process. This ID was sent in the *ArticleMasterSetRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *ArticleMasterSetResponse*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *ArticleMasterSetResponse*

Element	M/O	Data type	Description
SetResult	M	Tag	Result follows.
Attributes	**M/O**	**Data type**	**Description and Values**
Value	M	String	Displays the result of the master data process. Possible values: "Accepted" if the articles were accepted as a master article "Rejected" if the articles were not accepted as a master article.
Text	O	String	Any text for debugging and logging information. Can be used here for detailed error messages if the articles were not accepted.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <ArticleMasterSetResponse Id="1003" Source="999" Destination="100">
    <SetResult Value="Accepted" Text="Master Articles accepted."/>
  </ArticleMasterSetResponse>
</WWKS>
```

Library

See *ArticleMasterSetRequest*.

6.6.3 StockDeliverySetRequest

Structure

```
<WWKS>
  <StockDeliverySetRequest>
    <StockDelivery>
      <Article/>
    </StockDelivery>
  </StockDeliverySetRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockDeliverySetRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock delivery process. This ID is returned in the *StockDeliverySetResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *StockDeliverySetRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StockDeliverySetRequest*

Element	M/O	Data type	Description
StockDelivery	M	Tag	Stock delivery information follows. This element can be used multiple times.
DeliveryNumber	M	String	ID of the stock delivery. This must be a unique number within all active or pending stock deliveries. This number is used by the automated storage and retrieval system to assign packs placed into stock to a stock delivery.

Element	M/O	Data type	Description
Article	M	Tag	Article information follows. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the article. This may be the original barcode of the article or a derivative of it. The pharmacy IT system specifies how the article

			ID is composed. If the ID is a custom composition, the automated storage and retrieval system must be able to autonomously break the pack barcodes down into the relevant form.
Name	O	String	Name of the article
DosageForm	O	String	Dosage form of the article
PackingUnit	O	String	Packaging unit of the article
RequiresFridge	O	Boolean	Flag indicating whether the article has to be stored refrigerated ("True"). The default value is "False".
BatchNumber	O	String	Batch number which must be assigned to the packs of the article in this stock delivery
ExternalId	O	String	External ID which must be assigned to the packs of the article in this stock delivery
ExpiryDate	O	String	Expiry date in format YYYY-MM-DD which must be assigned to the packs of the article in this stock delivery
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which might be contained in a full pack of the article. The value "0" means that the number of units is unknown.
Quantity	O	Integer 32-bit >=0	Maximum number of packs of this article which may be placed into stock in this stock delivery. The value "0" means there is no limitation. The default value is "0".
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system that has to be used for packs of this article. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine that has to store packs of this article. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockDeliverySetRequest Id="1003" Source="100" Destination="999">
    <StockDelivery DeliveryNumber="1234">
      <Article Id="0004-56-034-G00007T" Name="ACCU CHEK AVIVA"
               DosageForm="LOE" PackagingUnit="1X2.5 ML"
               RequiresFridge="False" Quantity="15"/>
      <Article Id="56473627" Name="PREDNISOLO" DosageForm="TAB" PackagingUnit="20"
               BatchNumber="BAT918271" ExternalId="XT11725" ExpiryDate="2014-04-05"
               RequiresFridge="True" Quantity="5"/>
    </StockDelivery>
  </StockDeliverySetRequest>
</WWKS>
```

Library

If a pharmacy IT system is not permanently connected to the automated storage and retrieval system, or does not respond directly to stock input requests for other reasons, the pharmacy IT system can define stock deliveries with which the automated storage and retrieval system then works autonomously. Stock deliveries are created with the `CreateStockDelivery` method. Then the `AddItem` method is used to specify the articles in each stock delivery allowed to be placed into stock. Finally, the `AddStockDeliveries` method is used to transfer the stock delivery definitions to the automated storage and retrieval system.

```
var stockDeliveryList = new List<IStockDelivery>();

var stockDelivery = storageSystem.CreateStockDelivery("1234");
stockDelivery.AddItem("47463736",
                      "Article Number 1",
                      "Dosage Form",
                      "Packaging Unit");

stockDelivery.AddItem("78695739",
                      "Article Number 2",
                      "Dosage Form",
                      "Packaging Unit");

stockDeliveryList.Add(stockDelivery);

var stockDelivery2 = storageSystem.CreateStockDelivery("5132");
stockDelivery2.AddItem("68575484",
                       "Article Number 3",
                       "Dosage Form",
                       "Packaging Unit");

stockDelivery2.AddItem("69574362",
                       "Article Number 4",
                       "Dosage Form",
                       "Packaging Unit");

stockDeliveryList.Add(stockDelivery2);
storageSystem.AddStockDeliveries(stockDeliveryList);
```

6.6.4 StockDeliverySetResponse

Structure

```
<WWKS>
  <StockDeliverySetResponse>
    <SetResult/>
  </StockDeliverySetResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockDeliverySetResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock delivery process. This ID was sent in the *StockDeliverySetRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *StockDeliverySetResponse*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StockDeliverySetResponse*

Element	M/O	Data type	Description
SetResult	M	Tag	Result follows.
Attributes	**M/O**	**Data type**	**Description and Values**
Value	M	String	Displays the result of the stock delivery process. Possible values: "Accepted" if the predefined stock deliveries were accepted "Rejected" if the predefined stock deliveries were not accepted
Text	O	String	Any text for debugging and logging information. Can be used here for detailed error messages if the stock deliveries were not accepted.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockDeliverySetResponse Id="1003" Source="999" Destination="100">
    <SetResult Value="Accepted" Text="Stock Delivery accepted."/>
  </StockDeliverySetResponse>
</WWKS>
```

Library

See *StockDeliverySetRequest*.

6.7 Stock enquiry and stock change

Lead elements

StockInfoRequest

StockInfoResponse

StockInfoMessage

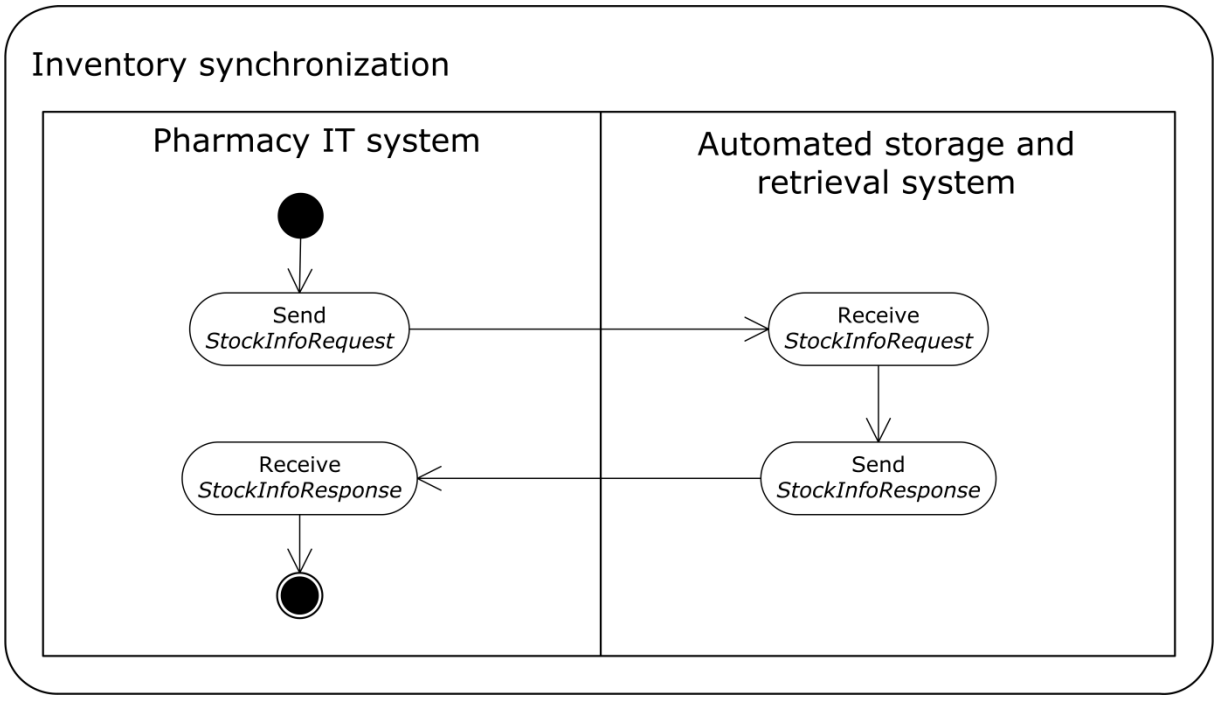
Usage

To poll the current stock level in the automated storage and retrieval system, the pharmacy IT system can send the *StockInfoRequest*. The automated storage and retrieval system will respond with *StockInfoResponse*.

The prompt can be constrained by filters. If filters are set, the automated storage and retrieval system's response only contains articles and packs matching the set criteria. If no filters are defined, the full inventory is listed.

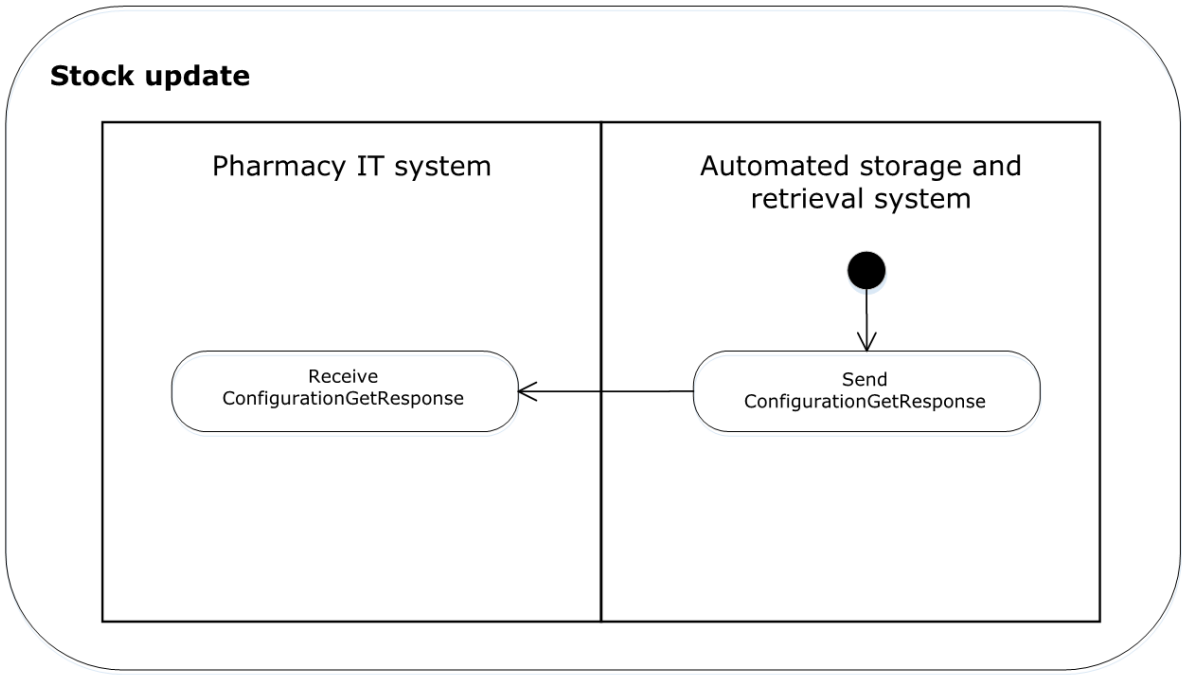
If there is no stock in the automated storage in retrieval system or if there are no articles corresponding to the filters defined in the search, the content of the *StockInfoResponse* message is empty (contains no *Article* elements).

Sequence



A stock update is a change of stock-related data (e.g. expiration date or status of a package). The number of stored items or packages does not change in the event of a stock update. The automated storage and retrieval system sends a message in the form of a StockInfoMessage to the pharmacy IT system.

Sequence



6.7.1 StockInfoRequest

Structure

```
<WWKS>
  <StockInfoRequest>
    <Criteria/>
  </StockInfoRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockInfoRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the inventory request. This ID is returned in the *StockInfoResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *StockInfoRequest*.
Destination	M	Integer 32-bit >0	ID of the system to which the *StockInfoRequest* is sent.
IncludePacks	O	Boolean	This flag specifies whether details of the existing packs are to be returned. Possible value: "True" if details of the packs are to be included "False" if only article data is to be sent The default value is "True".
IncludeArticleDetails	O	Boolean	This flag specifies whether detailed article information like name, dosage form, etc. are to be returned. Possible value: "True" if details of the articles are to be sent "False" if only minimal article data is to be sent The default value is "False".

Element	M/O	Data type	Description
Criteria	O	Tag	Request filter follows. Multiple criteria can be defined.
Attributes	**M/O**	**Data type**	**Description and Values**
ArticleId	O	String	By setting this filter, only articles with the specified article ID are included. The ID of the article must correspond to the one assigned by the pharmacy IT system when

			placing the article into stock in the *StockInputResponse*.
BatchNumber	O	String	By setting this filter, only packs with the specified batch number are included.
ExternalId	O	String	By setting this filter, only packs with the specified external ID are included.

StockLocationId	O	String	By setting this filter, only packs with the specified stock location ID within the automated storage and retrieval system are included. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	By setting this filter, only packs are with the specified identification of the machine used for storing the pack are included. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example without filter criteria

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockInfoRequest Id="1003" Source="100" Destination="999" />
</WWKS>
```

Example with filter criteria

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockInfoRequest Id="1003" Source="100" Destination="999" IncludePacks="True">
    <Criteria ArticleId="0004-56-034-G00007T"/>
    <Criteria BatchNumber="Omepra0004"/>
  </StockInfoRequest>
</WWKS>
```

Library

The inventory can be polled with the `GetStock` method:

```
List<IArticle> stockList = storageSystem.GetStock();
```

The full inventory is returned, including all pack details and all article details.

If, rather than pack details, only an article listing indicating the associated number of packs is needed, it is polled as follows:

```
List<IArticle> stockList = storageSystem.GetStock(false);
```

If only the stock of a specific article is needed, it is polled as follows:

```
List<IArticle> stockList = storageSystem.GetStock(true, "68575484");
```

6.7.2 StockInfoResponse

Structure

```
<WWKS>
  <StockInfoResponse>
    <Article>
      <Pack/>
    </Article>
  </StockInfoResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockInfoResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the inventory request. This ID was sent in the *StockInfoRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *StockInfoResponse*.
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StockInfoResponse*.

Element	M/O	Data type	Description
Article	M	Tag	Article information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the article. This ID corresponds to the one assigned by the pharmacy IT system when placing the article into stock in the *StockInputResponse*.
Name	O	String	Name of the article
DosageForm	O	String	Dosage form of the article
PackingUnit	O	String	Packaging unit of the article
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which might be contained in a full pack of the article. The value "0" means that the number of units is unknown.

Quantity	O	Integer 32-bit >0	Number of existing packs of this article

Element	**M/O**	**Data type**	**Description**
Pack	O	Tag	Pack information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	Integer 64-bit >0	Internal pack ID in the automated storage and retrieval system for a specific pack
DeliveryNumber	O	String	Stock delivery number specified on stock input
BatchNumber	O	String	Batch number. This was sent in the *InputResponse* during stock input.

ExternalId	O	String	External ID, an additional ID of the pack as sent in the *InputResponse* on stock input.
ExpiryDate	O	String	Expiry date of the pack in format YYYY-MM-DD
StockInDate	O	String	Input date of the pack in format YYYY-MM-DD
ScanCode	O	String	Barcode of the pack
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened).
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack. Possible value: "Cuboid" "Cylinder" The default value is "Cuboid".
State	O	String	Status of the pack. This data is required when multiple automatic storage machines are connected. Possible value: "Available" means that the pack is currently available for output. "NotAvailable" means that the pack is currently not available for output. The default value is "Available".
IsInFridge	O	Boolean	Flag indicating whether the pack is stored in a refrigeration unit ("True"). The default value is "False".
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockInfoResponse Id="1003" Source="999" Destination="100">
    <Article Id="0004-56-034-G00007T" Quantity="3">
      <Pack Id="4536" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" State="Available">
      <Pack Id="7664" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2012-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" State="Available">
      <Pack Id="7857" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2012-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" State="Available">
    </Article>
  </StockInfoResponse>
</WWKS>
```

Library

See *StockInfoRequest*.

6.7.3 StockInfoMessage

Structure

```
<WWKS>
  <StockInfoMessage>
    <Article>
      <Pack/>
    </Article>
  </StockInfoMessage>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockInfoMessage	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock update.
Source	M	Integer 32-bit >0	ID of the system sending the *StockInfoMessage*.
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StockInfoMessage.*

Element	M/O	Data type	Description
Article	M	Tag	Article information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the article. This ID corresponds to the one assigned by the pharmacy IT system when placing the article into stock in the *StockInputResponse*.
Name	O	String	Name of the article
DosageForm	O	String	Dosage form of the article
PackingUnit	O	String	Packaging unit of the article
MaxSubItemQuantity	O	Integer 32-bit >=0	Maximum number of units (e.g. pills or ampules) which might be contained in a full pack of the article. The value "0" means that the number of units is unknown.
Quantity	O	Integer 32-bit >0	Number of existing packs of this article

Element	M/O	Data type	Description
Pack	O	Tag	Pack information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	Integer 64-bit >0	Internal pack ID in the automated storage and retrieval system for a specific pack
DeliveryNumber	O	String	Stock delivery number specified on stock input
BatchNumber	O	String	Batch number. This was sent in the *InputResponse* during stock input.
ExternalId	O	String	External ID, an additional ID of the pack as sent in the *InputResponse* on stock input.
ExpiryDate	O	String	Expiry date of the pack in format YYYY-MM-DD
StockInDate	O	String	Input date of the pack in format YYYY-MM-DD
ScanCode	O	String	Barcode of the pack
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) currently inside the pack. The value "0" means that the pack is full (not opened).
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack. Possible value: "Cuboid" "Cylinder" The default value is "Cuboid".
State	O	String	Status of the pack. This data is required when multiple automatic storage machines are connected. Possible value: "Available" means that the pack is currently available for output. "NotAvailable" means that the pack is currently not available for output. The default value is "Available".

IsInFridge	O	Boolean	Flag indicating whether the pack is stored in a refrigeration unit ("True"). The default value is "False".
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockInfoMessage Id="1003" Source="999" Destination="100">
    <Article Id="0004-56-034-G00007T" Quantity="3">
      <Pack Id="4536" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" State="Available">
      <Pack Id="7664" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2012-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" State="Available">
      <Pack Id="7857" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2012-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" State="Available">
    </Article>
  </StockInfoMessage>
</WWKS>
```

Library

The event must be registered on `StockUpdated` to be notified about the stock updates:

```
storageSystem.StockUpdated += StorageSystem_StockUpdated;
```

If there is stock update, the automated storage and retrieval system sends a list of one or more items with the packages in question that have changed.

```
void StorageSystem_StockUpdated(IStorageSystem sender, IArticle[] articleList)
{
    foreach (var article in articleList)
    {
        for each (var pack in article.Packs)
        {
            // process pack related updates
        }
    }
}
```

6.8 Stock output

Lead elements

OutputRequest

OutputResponse

OutputMessage

Usage

When the pharmacy IT system wants packs withdrawn from stock, it sends the *OutputRequest* to the automated storage and retrieval system. The automated storage and retrieval system responds in the *OutputResponse* indicating whether it is able to execute the output or not. The automated storage and retrieval system may refuse an output request if, for example, the system is not ready due to maintenance procedures or because the query contains invalid data. If the automated storage and retrieval system accepts the output request, the output is executed as quickly as possible. After output (but also if the output fails) the automated storage and retrieval system sends the *OutputMessage*.

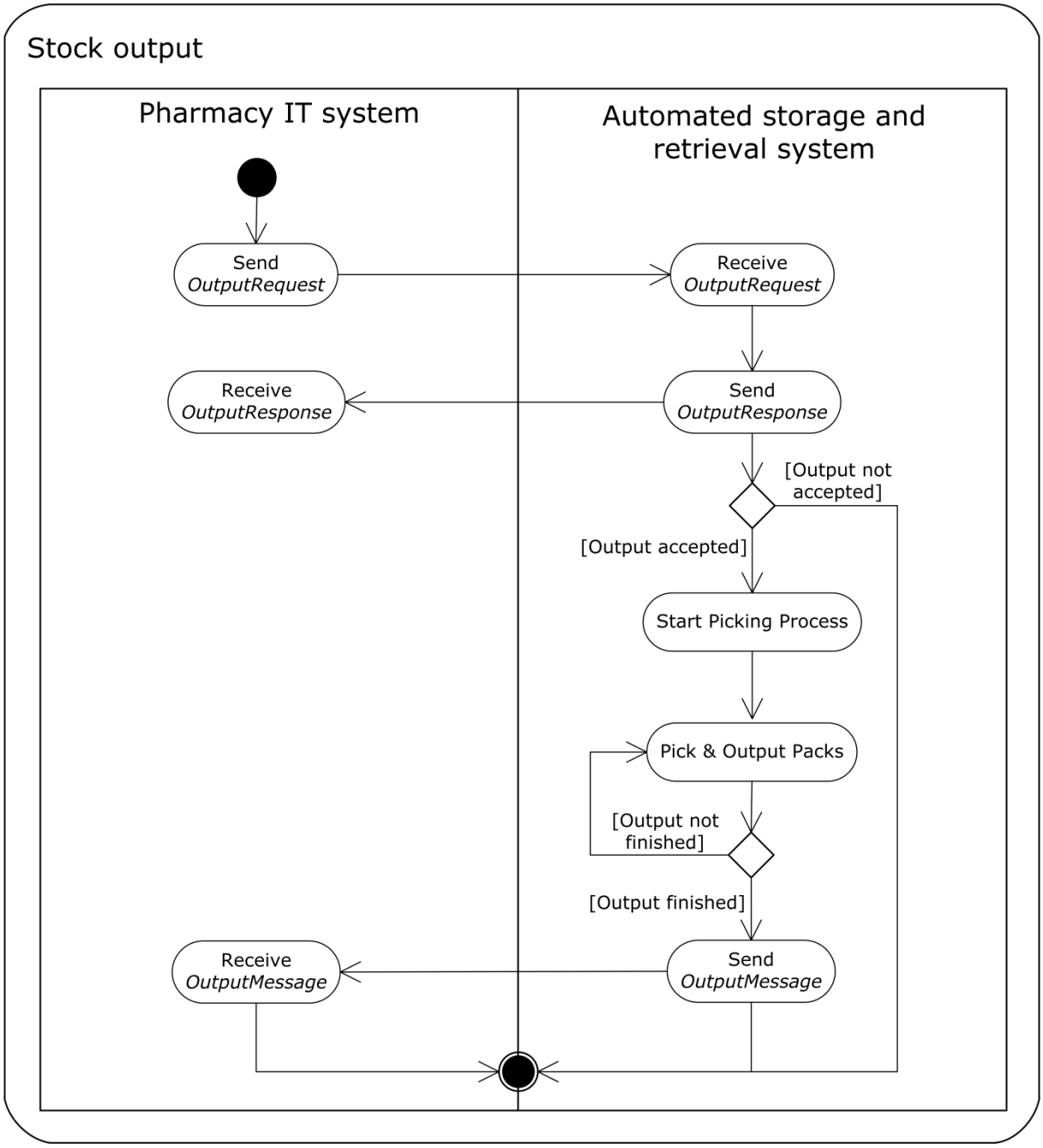
The pharmacy IT system can send multiple output requests in sequence. The automated storage and retrieval system will process them in the order in which they are received or according to priority, as appropriate.

Some automated storage and retrieval systems offer the facility to request packs directly on the automated storage and retrieval system without going through the pharmacy IT system. This is done by an operative on the automated storage and retrieval system's user interface. This procedure requires a returned *OutputMessage* from the automated storage and retrieval system to the pharmacy IT system. You will find an example of this so-called manual output at the end of this section.

Some automated storage and retrieval systems support labeling of packs during the stock output process. A special printer is connected to the automated storage and retrieval system for the purpose. The pharmacy IT system can define the content of the printed label in the *OutputRequest*. The label print is configured in detail both by the pharmacy IT system and on the label printer. Templates are used to specify how the data is presented. The data sent to the label printer may be in a variety of formats, such as XML, HTML or SVG, depending on what the printer is able to process. The label content is therefore embedded in the *OutputRequest* and *OutputResponse* as a CDATA element.

In combination with automated container filling systems, it is possible that the automated storage and retrieval system sends a so-called empty-order to the pharmacy IT system. An empty-order does not contain any *Criteria* elements and therefore does not prompt a pack output. Consequently the affected container is forwarded by the automated storage and retrieval system as soon as it reaches the container filling point of the automated storage and retrieval system. If no container number is specified in an empty-order, the next container reaching the container filling point of the automated storage and retrieval system is forwarded and the corresponding container number is returned to the pharmacy IT system in an *OutputMessage*.

Sequence


Stock output
Pharmacy IT system
Automated storage and retrieval system
Send OutputRequest
Receive OutputRequest
Receive OutputResponse
Send OutputResponse
[Output not accepted]
[Output accepted]
Start Picking Process
Pick & Output Packs
[Output not finished]
[Output finished]
Receive OutputMessage
Send OutputMessage

6.8.1 OutputRequest

Structure

```
<WWKS>
  <OutputRequest>
    <Details/>
    <Criteria>
      <Label>
        <Content>
        </Content>
      </Label>
    </Criteria>
  </OutputRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
OutputRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock output process. This ID is returned in the *OutputResponse* and is also used in the associated *OutputMessage*.
Source	M	Integer 32-bit >0	ID of the system sending the output order
Destination	M	Integer 32-bit >0	ID of the system receiving the output order
BoxNumber	O	String	Number of the box to be used for this task. The data is only required if an automatic box filler is being used and the pharmacy IT system predefines the box numbers for an output order.

Element	M/O	Data type	Description
Details	M	Tag	Output details follow.
Attributes	**M/O**	**Data type**	**Description and Values**
Priority	O	String	Priority of the stock output. Possible values: "Low" "Normal" "High" The default value is "Normal".

OutputDestination	M	Integer 32-bit	Number of the output location to which the packs are to be sent
OutputPoint	O	Integer 32-bit	Detailed information on the requested output location (e.g. belt number).

Element	**M/O**	**Data type**	**Description**
Criteria	O	Tag	Output filters follow. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
ArticleId	O	String	The article ID serving as a filter for the packs. The ID must exactly match the one in the *StockInputResponse*.
Quantity	M	Integer 32-bit >=0	Number of full packs to be outputted
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets, ampules) to be outputted. The automated storage and retrieval system calculates the resultant number of packs and outputs them. If this attribute is used, the *Quantity* attribute is ignored. It should nevertheless be set to "0".
MinimumExpiryDate	O	String	Filter for packs having the specified expiry date as a minimum. Format YYYY-MM-DD.
BatchNumber	O	String	Filter for packs having the specified batch number
SingleBatchNumber	O	Boolean	Alternative to BatchNumber. Defines that all requested packs should have the same BatchNumber. The default value is False.
ExternalId	O	String	Filter for packs having the specified external ID
PackId	O	Integer 64-bit >0	Filter for packs having the specified automated storage and retrieval system internal pack ID. This filter can be used to output a specific pack.
StockLocationId	O	String	Filter for packs having the specified ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Filter for packs having the specified identification of the machine used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Element	M/O	Data type	Description
Label	O	Tag	Label information follows. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
TemplateId	M	String	ID of the label template to be used by the label printer for correct output of the label content

Element	M/O	Data type	Description
Content	M	Tag	Label content to be printed. This data is embedded in the message as a CDATA XML block.

Example of an empty output request

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputRequest Id="1004" Source="100" Destination="999">
    <Details Priority="Normal" OutputDestination="3"/>
  </OutputRequest>
</WWKS>
```

Example of an output request with article ID

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputRequest Id="1004" Source="100" Destination="999">
    <Details Priority="Normal" OutputDestination="3"/>
    <Criteria ArticleId="0004-56-034-G00025T" Quantity="1"/>
    <Criteria ArticleId="0004-56-034-G00007T" Quantity="1"
              MinimumExpiryDate="2015-11-01"/>
  </OutputRequest>
</WWKS>
```

Example of an output request with article ID and external ID

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputRequest Id="1004" Source="100" Destination="999">
    <Details Priority="Normal" OutputDestination="3"/>
    <Criteria ArticleId="0004-56-034-G00007T"
              ExternalId="PalH09051200001" Quantity="1"/>
  </OutputRequest>
</WWKS>
```

Example of an output request with article ID and label data

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputRequest Id="1004" Source="100" Destination="999">
    <Details Priority="Normal" OutputDestination="3"/>
    <Criteria ArticleId="0004-56-034-G00007T" Quantity="1">
      <Label TemplateId="3413">
        <Content>
          <![CDATA[
            <article>
              <name>NIFEDIPIN 20 retard 1A Pharma Tabl.</name>
              <quantity>30</quantity>
            </article>
            <dosagelines>
              <labeldosageline>
                <synonym>on an empty stomach</synonym>
                <amount>1</amount>
                <quantityunit>pcs</quantityunit>
              </labeldosageline>
            </dosagelines>
          ]]>
        </Content>
      </Label>
    </Criteria>
  </OutputRequest>
</WWKS>
```

Library

The `CreateOutputProcess` method is used to create an output process.
Then the `AddCriteria` method is used to define the output criteria
(e.g. *n* packs of article *x*).
The `Start` method is called to send the output order to the automated storage and retrieval system.

In the following example, an output order with the number *1234* is created, requesting a stock output to point *2*. The output is to include one pack of article *47463736* and two packs of article *78695739*.

```
IOutputProcess output = storageSystem.CreateOutputProcess(1234, 2);
output.AddCriteria("47463736", 1);
output.AddCriteria("78695739", 2);

// optionally add label content to a criteria
output.Criteria[0].AddLabel("4536", "<article>…");

output.Start();
```

6.8.2 OutputResponse

Structure

```
<WWKS>
  <OutputResponse>
    <Details/>
    <Criteria>
      <Label>
        <Content>
        </Content>
      </Label>
    </Criteria>
  </OutputResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
OutputResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock output process. This ID was sent in the *OutputRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *OutputResponse*
Destination	M	Integer 32-bit >0	ID of the system receiving the *OutputResponse*
BoxNumber	O	String	Number of the box specified in the *OutputRequest*. The data is required if an automatic box filler is being used.

Element	M/O	Data type	Description
Details	M	Tag	Output details follow.
Attributes	**M/O**	**Data type**	**Description and Values**
Priority	O	String	Priority of this stock output process. Here the same value is to be seen as in the *OutputRequest*.
OutputDestination	M	Integer 32-bit	Number of the output location to which the packs are sent. Here the same value is to be seen as in the *OutputRequest*.

OutputPoint	O	Integer 32-bit	Detailed information on the requested output location (e.g. belt number). Here the same value is to be seen as in the *OutputRequest*.
Status	M	String	Status of the stock output process. Possible values: "Queued" if the output is pending "Rejected" if the output is rejected

Element	**M/O**	**Data type**	**Description**
Criteria	O	Tag	Output filters follow. This element may occur multiply. The attributes and values correspond to those in the *OutputRequest*.
Attributes	**M/O**	**Data type**	**Description and Values**
ArticleId	O	String	The article ID serving as a filter for the packs. The ID must exactly match the one in the *StockInputResponse*.
Quantity	M	Integer 32-bit >=0	Number of full packs to output. Here the same value is to be seen as in the *OutputRequest*.
SubItemQuantity	O	String	Number of units (e.g. tablets, ampules) to be outputted. If this attribute is used, the *Quantity* attribute is ignored. It should nevertheless be set to "0". Here the same value is to be seen as in the *OutputRequest*.
MinimumExpiryDate	O	Integer 32-bit >=0	Filter for packs having the specified expiry date as a minimum. Format YYYY-MM-DD. Here the same value is to be seen as in the *OutputRequest*.
BatchNumber	O	String	Filter for packs having the specified batch number. Here the same value is to be seen as in the *OutputRequest*.
SingleBatchNumber	O	Boolean	Alternative to BatchNumber. Defines that all requested packs should have the same BatchNumber. The default value is False. Here the same value is to be seen as in the *OutputRequest*.
ExternalId	O	String	Filter for packs having the specified external ID. Here the same value is to be seen as in the *OutputRequest*.
PackId	O	Integer 64-bit >0	Filter for packs having the specified automated storage and retrieval system internal pack ID. This filter can be used to output a specific pack. Here the same value is to be seen as in the *OutputRequest*.

StockLocationId	O	String	Filter for packs having the specified ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations. Here the same value is to be seen as in the *OutputRequest*.
MachineLocation	O	String	Filter for packs having the specified identification of the machine used for storing the pack. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines. Here the same value is to be seen as in the *OutputRequest*.

Element	**M/O**	**Data type**	**Description**
Label	O		Label information follows. This element may occur multiply. The data corresponds to that in the *OutputRequest*.
Attributes	**M/O**	**Data type**	**Description and Values**
TemplateId	M		ID of the label template to be used by the label printer for correct output of the label content. The data corresponds to that in the *OutputRequest*.

Element	**M/O**	**Data type**	**Description**
Content	M		Label content to be printed. This data is embedded in the message as a CDATA XML block. The data corresponds to that in the *OutputRequest*.

Example of an accepted output request

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputResponse Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Queued"/>
    <Criteria ArticleId="0004-56-034-G00025T" Quantity="1"/>
    <Criteria ArticleId="0004-56-034-G00007T" Quantity="1"
              MinimumExpiryDate="2015-11-01"/>
  </OutputResponse>
</WWKS>
```

Example of a rejected output request

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputResponse Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Rejected"/>
    <Criteria ArticleId="0004-56-034-G00025T" Quantity="1"/>
    <Criteria ArticleId="0004-56-034-G00007T" Quantity="1"
              MinimumExpiryDate="2015-11-01"/>
  </OutputResponse>
</WWKS>
```

Example of an accepted output request with label data

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputResponse Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Queued"/>
    <Criteria ArticleId="0004-56-034-G00007T" Quantity="1">
      <Label TemplateId="3413">
        <Content>
          <![CDATA[
            <article>
              <name>NIFEDIPIN 20 retard 1A Pharma Tabl.</name>
              <quantity>30</quantity>
            </article>
            <dosagelines>
              <labeldosageline>
                <synonym>on an empty stomach</synonym>
                <amount>1</amount>
                <quantityunit>pcs</quantityunit>
              </labeldosageline>
            </dosagelines>
          ]]>
        </Content>
      </Label>
    </Criteria>
  </OutputResponse>
</WWKS>
```

Library

There are two ways of waiting for the task to complete.

1st option: Check the status of the task regularly in a loop.

```
while (true)
{
       OutputProcessState currentState = output.State;
       if ((currentState != OutputProcessState.Queued) &&
           (currentState != OutputProcessState.InProcess))
       {
             break;
       }
       Thread.Sleep(1000);
}
```

2nd option (recommended): Register for the `Finished` event before calling the `Start`method.

```
output.Finished += OutputProcess_Finished;
output.Start();

void OutputProcess_Finished(object sender, EventArgs e)
{
       var output = sender as IOutputProcess;
       Console.WriteLine("Output process '{0}' finished with result '{1}'.",
                         output.OrderNumber, output.State.ToString());
}
```

6.8.3 OutputMessage

Structure

```
<WWKS>
  <OutputMessage>
    <Details/>
    <Article>
      <Pack/>
    </Article>
    <Box/>
  </OutputMessage>
</WWKS>
```

Elements

Element	M/O	Data type	Description
OutputMessage	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock output process. This ID was sent in the *OutputRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *OutputMessage*
Destination	M	Integer 32-bit >0	ID of the system receiving the *OutputMessage*

Element	M/O	Data type	Description
Details	M	Tag	Output details follow.
Attributes	**M/O**	**Data type**	**Description and Values**
Priority	O	String	Priority of this stock output process. Here the same value is to be seen as in the *OutputRequest*.
OutputDestination	M	Integer 32-bit	Number of the output location to which the packs were sent. Here the same value is to be seen as in the *OutputRequest*.
OutputPoint	O	Integer 32-bit	Detailed information on the requested output location (e.g. belt number). Here the same value is to be seen as in the *OutputRequest*.
Status	M	String	Status of the stock output process. Possible values: "Completed" if the output was completed successfully

			"Incomplete" if the output was not completed fully "Aborted" if the output was aborted

Element	**M/O**	**Data type**	**Description**
Article	O	Tag	Article information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	O	String	Article ID of the outputted pack

Element	**M/O**	**Data type**	**Description**
Pack	O	Tag	Pack information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	Integer 64-bit >0	Picking system internal ID of the outputted pack
DeliveryNumber	O	String	Stock delivery number of the outputted pack
BatchNumber	O	String	Batch number of the outputted pack
ExternalId	O	String	External ID of the outputted pack
ExpiryDate	O	String	Expiry date of the outputted pack in format YYYY-MM-DD.
StockInDate	O	String	Input date of the output pack in format YYYY-MM-DD
ScanCode	O	String	Barcode of the output pack
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) in the outputted pack. The value "0" means that the pack is full.
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack.

			Possible value: "Cuboid" "Cylinder" The default value is "Cuboid".
IsInFridge	O	Boolean	Flag indicating whether the pack has been stored refrigerated. The default value is "False".
BoxNumber	O	String	Number of the box to which the pack was outputted. The data is only required if an automatic box filler is being used.
OutputDestination	M	Integer 32-bit	Number of the output location to which the pack was sent. A possible error case is output to a point other than the one requested, for example because the requested output point was not operational.
OutputPoint	O	Integer 32-bit	Detailed information on the output location (e.g. belt number) used for pack output.
LabelStatus	O	String	Status of the labeling of the outputted pack. This is only relevant in conjunction with labeling. Possible values: "Labelled" if the pack was labeled correctly with the data predefined by the pharmacy IT system "NotLabelled" if the pack was not labeled (e.g. because no printer was available) "LabelError" if an error occurred during labeling
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine that was used to store packs of this article. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Element	**M/O**	**Data type**	**Description**
Box	O	Tag	Detailed information on the containers used follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Number	M	String	Number of the container either defined in the *OutputRequest* or automatically determined during filling.

Example of a successfully completed stock output

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputMessage Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Completed"/>
    <Article Id="0004-56-034-G00025T">
      <Pack Id="5637" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" OutputDestination="3" LabelStatus="Labelled"/>
    </Article>
    <Article Id="0004-56-034-G00007T">
      <Pack Id="8563" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="70" Width="70" Height="70"
            Shape="Cuboid" IsInFridge="True" OutputDestination="3"
            LabelStatus="Labelled"/>
    </Article>
  </OutputMessage>
</WWKS>
```

Example of a partially completed stock output (not enough packs available)

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputMessage Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Incomplete"/>
    <Article Id="0004-56-034-G00025T">
      <Pack Id="5637" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" OutputDestination="3"/>
    </Article>
  </OutputMessage>
</WWKS>
```

Example of a stock output canceled before output began

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputMessage Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Aborted"/>
  </OutputMessage>
</WWKS>
```

Example of a stock output aborted after output began

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputMessage Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Aborted"/>
    <Article Id="0004-56-034-G00025T">
      <Pack Id="5637" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" OutputDestination="3"/>
    </Article>
  </OutputMessage>
</WWKS>
```

Example of a successfully completed stock output with container filling

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputMessage Id="1004" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Completed"/>
    <Article Id="0004-56-034-G00025T">
      <Pack Id="5637" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" OutputDestination="3" LabelStatus="Labelled"
            BoxNumber="123" />
    </Article>
    <Article Id="0004-56-034-G00007T">
      <Pack Id="8563" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="70" Width="70" Height="70"
            Shape="Cuboid" IsInFridge="True" OutputDestination="3"
            LabelStatus="Labelled" BoxNumber="456" />
    </Article>
    <Box Number="123" />
    <Box Number="456" />
  </OutputMessage>
</WWKS>
```

Library

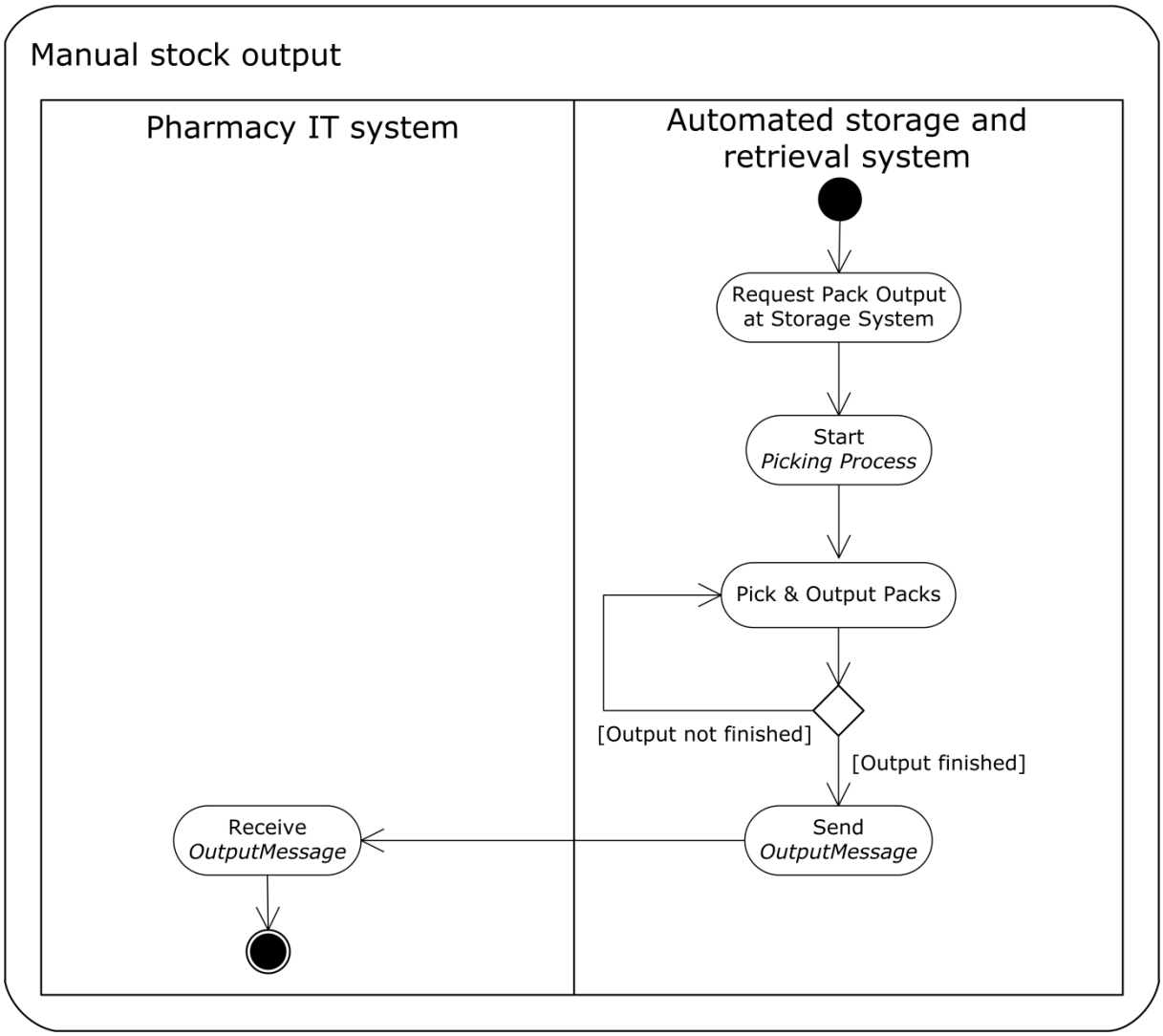
As soon as an output order has been completed, the final process status and a detailed list of outputted packs can be polled:

```
void OutputProcess_Finished(object sender, EventArgs e)
{
       var output = sender as IOutputProcess;
       foreach (var pack in output.Packs)
       {
              Console.WriteLine("Pack '{0}' was dispensed by output process {1}.",
                                  pack.Id, output.OrderNumber);
       }
}
```

Manual stock output

Some automated storage and retrieval systems offer the facility to request packs directly on the automated storage and retrieval system without going through the pharmacy IT system. This is done by an operative on the automated storage and retrieval system's user interface. After output, an asynchronous *OutputMessage* is returned from the automated storage and retrieval system to the pharmacy IT system. The ID of this "manual" output process is always "1".

Sequence



Example of a successfully completed manual stock output

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <OutputMessage Id="1" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="3" Status="Completed"/>
    <Article Id="0004-56-034-G00025T">
      <Pack Id="5637" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
            ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
            Shape="Cuboid" OutputDestination="3"/>
    </Article>
  </OutputMessage>
</WWKS>
```

Library

To be able to respond to manually outputted packs, the user must register for the `PackDispensed` event. This event is called whenever a pack has been outputted without an output order.

```
storageSystem.PackDispensed += StorageSystem_PackDispensed;

void StorageSystem_PackDispensed(IStorageSystem sender, IArticle[] articleList)
{
    foreach (var article in articleList)
    {
        foreach (var pack in article.Packs)
        {
            Console.WriteLine("Pack '{0}' has been dispensed by GUI.", pack.Id);
        }
    }
}
```

6.9 Task status

Lead elements

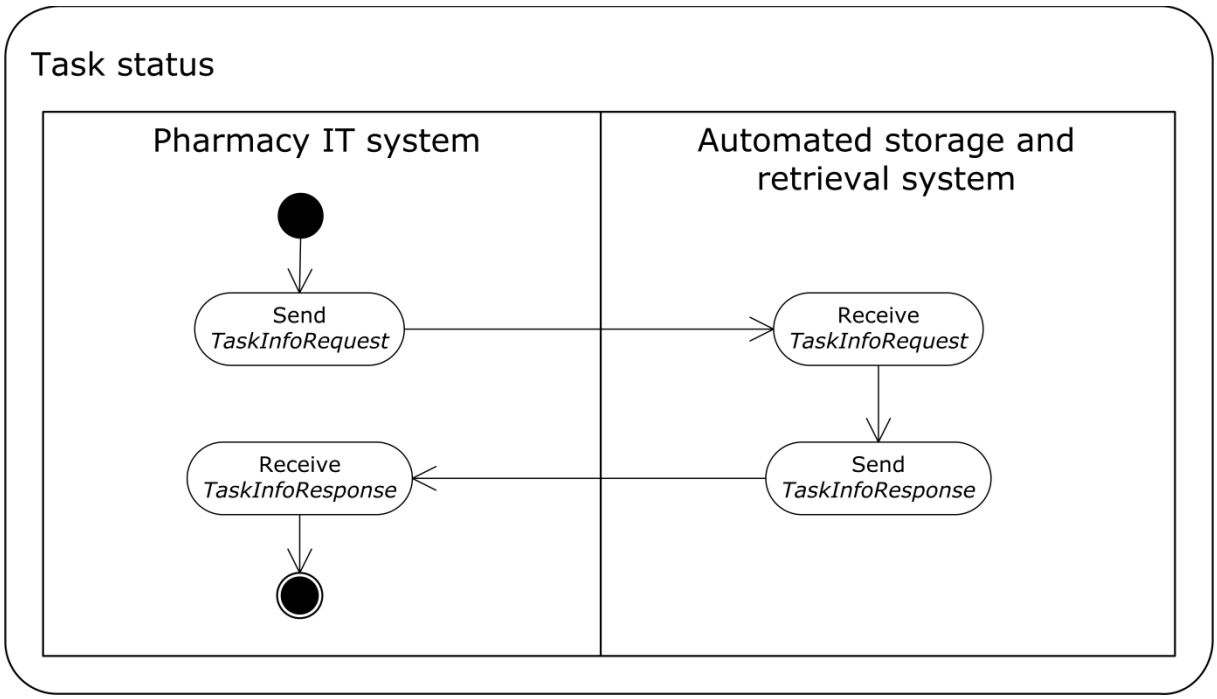
TaskInfoRequest

TaskInfoResponse

Usage

To poll the current status of a task (such as editing a stock output or a new delivery) in progress on the automated storage and retrieval system, the pharmacy IT system can send the *TaskInfoRequest*. The automated storage and retrieval system will respond to it with the *TaskInfoResponse*.

Sequence



6.9.1 TaskInfoRequest

Structure

```
<WWKS>
  <TaskInfoRequest>
    <Task/>
  </TaskInfoRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
TaskInfoRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the information process. This ID is returned in the *TaskInfoResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *TaskInfoRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *TaskInfoRequest*
IncludeTaskDetails	O	Boolean	This flag specifies whether detailed task information like outputted packs, etc. are to be returned. Possible value: "True" if details of the tasks are to be sent "False" if no task data is to be sent The default value is "False".

Element	M/O	Data type	Description
Task		Tag	Task information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
Type	M	String	Type of task on which information is being polled. Possible value: "Output" and "StockDelivery"
Id	M	String	ID of the task. For output tasks, this is the ID specified in the *OutputRequest*. For new deliveries, this is the DeliveryNumber specified in the *StockDeliverySetRequest*.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskInfoRequest Id="3330" Source="100" Destination="999">
    <Task Type="Output" Id="1004"/>
  </TaskInfoRequest>
</WWKS>
```

Library

As long as an output task is in progress, every time the attribute`State` is accessed the current status of the output order is determined.

Alternatively, it is possible to check the detailed information on running or already completed output orders via the `GetOutputProcessInfo` method.

```
// retrieve the detailed information of output process 124
IOutputProcessInfo info = storageSystem.GetOutputProcessInfo(124);
```

Detailed information on a running or an already completed new delivery can be checked via the `GetStockDeliveryInfo` method.

```
// retrieve the detailed information of stock delivery 1234
IStockDeliveryInfo info = storageSystem.GetStockDeliveryInfo("1234");
```

6.9.2 TaskInfoResponse

Structure

```
<WWKS>
  <TaskInfoResponse>
    <Task>
      <Article>
        <Pack/>
      </Article>
      <Box/>
    </Task>
  </TaskInfoResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
TaskInfoResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the information process. This ID was sent in the *TaskInfoRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *TaskInfoResponse*.

Destination	M	Integer 32-bit >0	ID of the system intended to receive the *TaskInfoResponse*.

Element	**M/O**	**Data type**	**Description**
Task	M	Tag	Task information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
Type	M	String	Type of task on which information is being polled. Possible value: "Output" and "StockDelivery"

Id	M	String	ID of the task. For output tasks, this is the ID specified in the *OutputRequest*. For new deliveries, this is the DeliveryNumber specified in the *StockDeliverySetRequest*.
Status	M	String	Status of the task. Possible value: "Unknown" if the task was not found "Queued" if processing is pending "InProcess" if currently being processed "Aborting" if the process is currently being aborted "Aborted" if processing has been aborted "Completed" if the task is complete "Incomplete" if the task was not completed fully

Element	**M/O**	**Data type**	**Description**
Article	O	Tag	Article information follows.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	O	String	Article ID of the affected pack
Quantity	O	Integer 32-bit >=0	This data is only used for the "StockDelivery" task type. Maximum number of packs of this article which may be placed into stock in this stock delivery. The value "0" means there is no limitation. The default value is "0".

Element	M/O	Data type	Description
Pack	O	Tag	Pack information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	Integer 64-bit >0	Picking system internal ID of the affected pack
DeliveryNumber	O	String	Stock delivery number of the affected pack
BatchNumber	O	String	Batch number of the affected pack
ExternalId	O	String	External ID of the affected pack
ExpiryDate	O	String	Expiry date of the affected pack in format YYYY-MM-DD.
StockInDate	O	String	Input date of the output pack in format YYYY-MM-DD
ScanCode	O	String	Barcode of the output pack
SubItemQuantity	O	Integer 32-bit >=0	Number of units (e.g. tablets or ampules) in the affected pack. The value "0" means that the pack is full.
Depth	O	Integer 32-bit >=0	Depth of the pack in mm
Width	O	Integer 32-bit >=0	Width of the pack in mm
Height	O	Integer 32-bit >=0	Height of the pack in mm
Shape	O	String	Form factor of the pack. Possible value: "Cuboid" "Cylinder" The default value is "Cuboid".
IsInFridge	O	Boolean	Flag indicating whether the pack has been or is being stored refrigerated. The default value is "False".
BoxNumber	O	String	Number of the box to which the pack was outputted. The data is only required for the "Output" task type if an automatic box filler is being used.
OutputDestination	M	Integer 32-bit	This data is only required for the "Output" task type.

			Number of the output location to which the pack was sent. A possible error case is output to a point other than the one requested, for example because the requested output point was not operational.
OutputPoint	O	Integer 32-bit	This data is only required for the "Output" task type. Detailed information on the output location (e.g. belt number) used for pack output.
LabelStatus	O	String	This data is only required for the "Output" task type. Status of the labeling of the outputted pack. This is only relevant in conjunction with labeling. Possible values: "Labelled" if the pack was labeled correctly with the data predefined by the pharmacy IT system "NotLabelled" if the pack was not labeled (e.g. because no printer was available) "LabelError" if an error occurred during labeling
StockLocationId	O	String	ID of the stock location in the automated storage and retrieval system. Is only used when an automated storage and retrieval system is divided into several virtual stock locations.
MachineLocation	O	String	Identification of the machine that was or will be used to store packs of this article. Only relevant if the automated storage and retrieval system consists of several physical stand-alone machines.

Element	**M/O**	**Data type**	**Description**
Box	O	Tag	This data is only required for the "Output" task type. Detailed information on the containers used follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Number	M	String	Number of the container either defined in the *OutputRequest* or automatically determined during filling.

Example for Output without details

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskInfoResponse Id="3330" Source="999" Destination="100">
    <Task Type="Output" Id="1004" Status="InProcess"/>
  </TaskInfoResponse>
</WWKS>
```

Example for Output with details

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskInfoResponse Id="3330" Source="999" Destination="100">
    <Task Type="Output" Id="1004" Status="Completed">
      <Article Id="0004-56-034-G00025T">
        <Pack Id="5637" BatchNumber="Omepra0004" ExternalId="PalH09051200001"
              ExpiryDate="2015-11-05" Depth="50" Width="50" Height="50"
              Shape="Cuboid" OutputDestination="3" LabelStatus="Labelled"
              BoxNumber="123" />
      </Article>
      <Box Number="123" />
    </Task>
  </TaskInfoResponse>
</WWKS>
```

Example for StockDelivery without details

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskInfoResponse Id="3330" Source="999" Destination="100">
    <Task Type="StockDelivery" Id="1234" Status="InProcess"/>
  </TaskInfoResponse>
</WWKS>
```

Example for StockDelivery with details

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskInfoResponse Id="3330" Source="999" Destination="100">
    <Task Type="StockDelivery" Id="1234" Status="Completed">
      <Article Id="0004-56-034-G00025T" Quantity="1">
        <Pack Id="5637" DeliveryNumber="1234" BatchNumber="Omepra0004"
              ExternalId="PalH09051200001" ExpiryDate="2015-11-05"
              Depth="50" Width="50" Height="50" Shape="Cuboid" />
      </Article>
    </Task>
  </TaskInfoResponse>
</WWKS>
```

Library

See *TaskInfoRequest*.

6.10 Cancellation

Lead elements

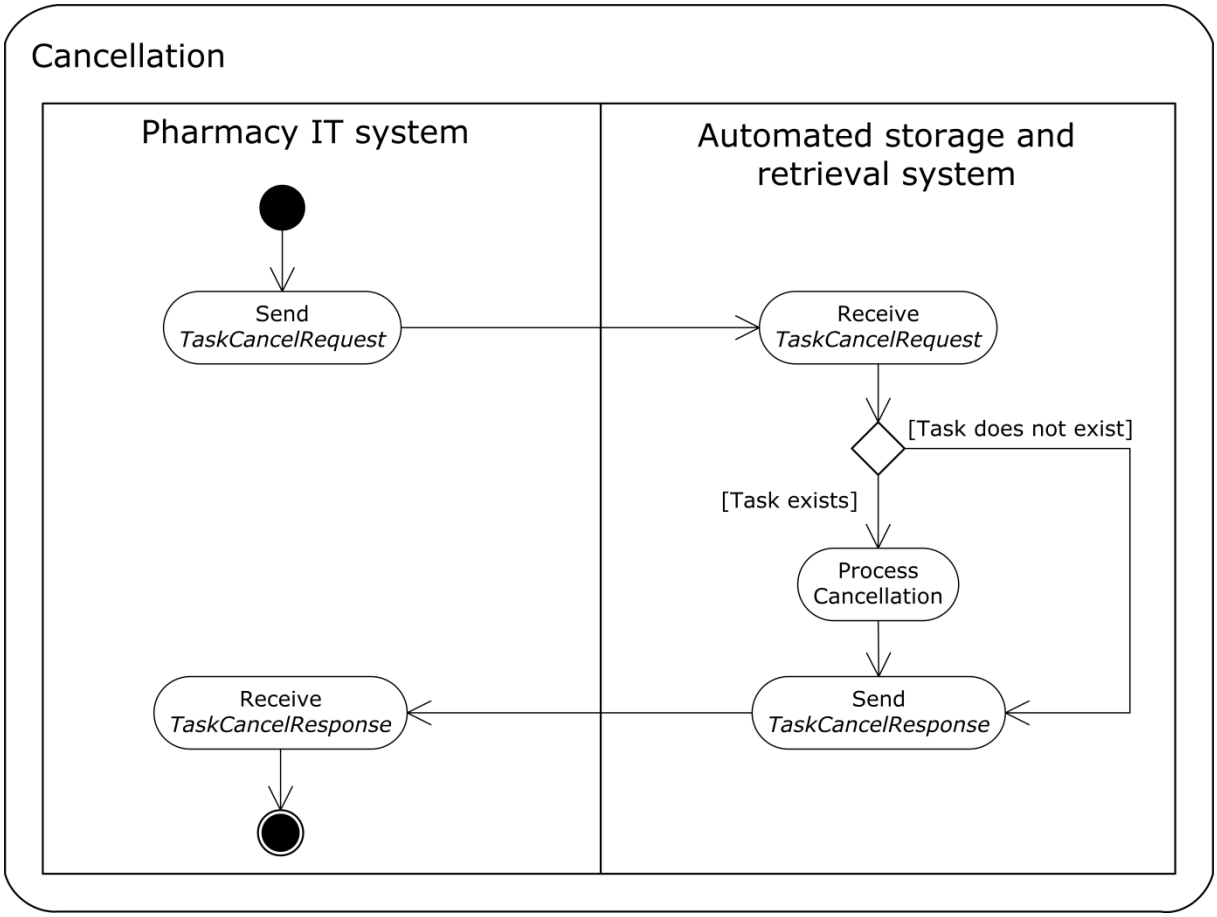
TaskCancelRequest

TaskCancelResponse

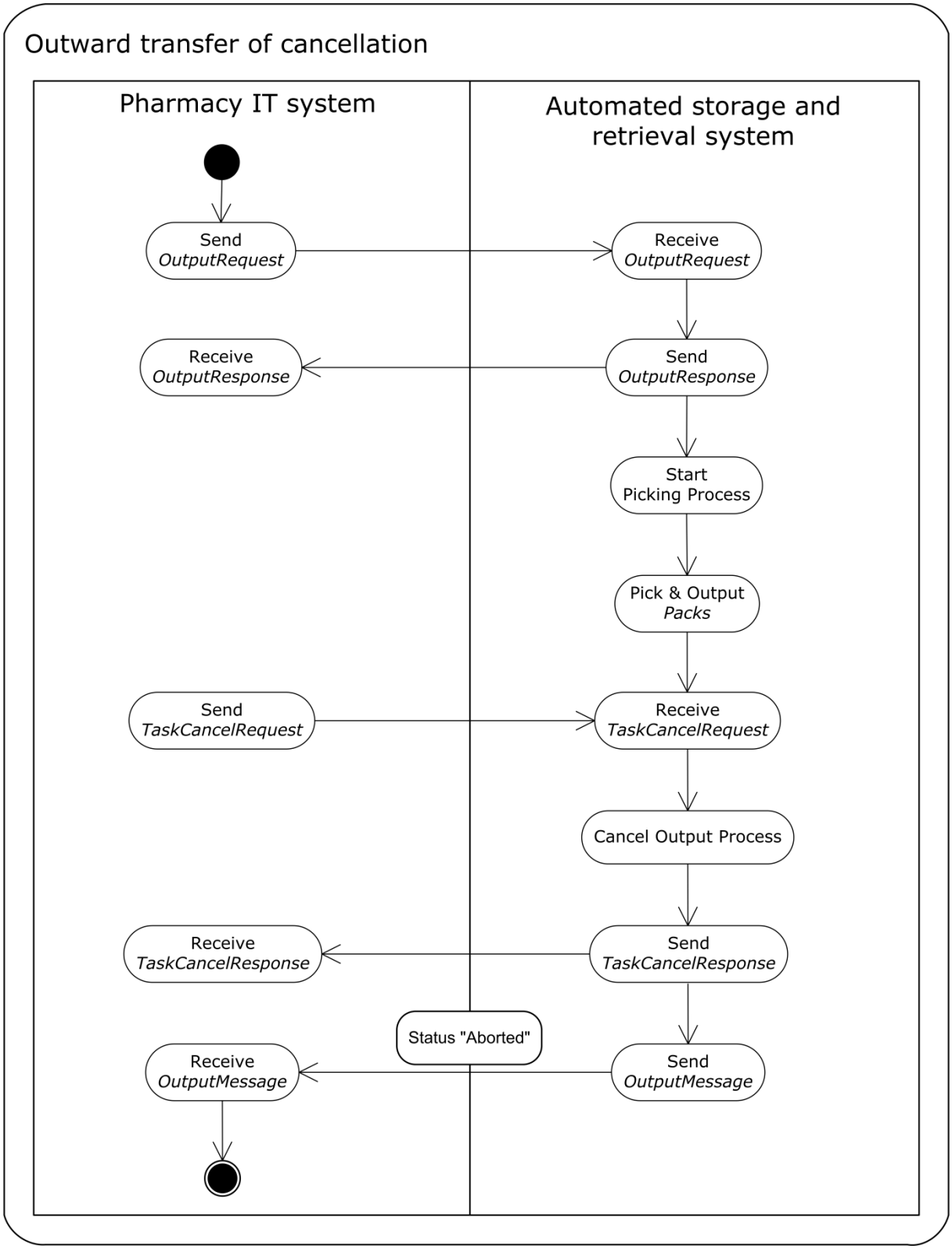
Usage

To cancel a task (such as a stock output) in progress on the automated storage and retrieval system, the pharmacy IT system can send the *TaskCancelRequest*. The automated storage and retrieval system will cancel processing, if possible, and send the *TaskCancelResponse*.

Sequence



The following diagram shows the cancellation function in a stock output process.



6.10.1 TaskCancelRequest

Structure

```
<WWKS>
  <TaskCancelRequest>
    <Task/>
  </TaskCancelRequest>
</WWKS>
```

Elements

Element	M/O	Data type	Description
TaskCancelRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the cancelation process. This ID is returned in the *TaskCancelResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *TaskCancelRequest*
Destination	M	Integer 32-bit >0	ID of the automated storage and retrieval system intended to receive the *TaskCancelRequest*

Element	M/O	Data type	Description
Task	M	Tag	Task information follows. This element can be used multiple times.
Attributes	**M/O**	**Data type**	**Description and Values**
Type	M	String	Type of task to be canceled. Possible value: "Output"
Id	M	String	ID of the task to be canceled. For output tasks, this is the ID specified in the *OutputRequest*.

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskCancelRequest Id="3335" Source="100" Destination="999">
    <Task Type="Output" Id="1004"/>
  </TaskCancelRequest>
</WWKS>
```

Library

An ongoing stock output order can be canceled by the `Cancel` method applied to an `IOutputProcess` object:

```
output.Cancel();
```

6.10.2 TaskCancelResponse

Structure

```
<WWKS>
  <TaskCancelResponse>
    <Task/>
  </TaskCancelResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
TaskCancelResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the cancelation process. This ID was sent in the *TaskCancelRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *TaskCancelResponse*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *TaskCancelResponse*

Element	M/O	Data type	Description
Task	M	Tag	Task information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Type	M	String	Type of task canceled. Possible value: "Output"
Id	M	String	ID of the canceled task. For output tasks, this is the ID specified in the *OutputRequest*.
Status	M	String	Status of the cancellation. Possible value: "Unknown" if the task is not known "Cancelled" if the task could be canceled "CancelError", if the cancellation failed

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <TaskCancelResponse Id="3335" Source="999" Destination="100">
    <Task Type="Output" Id="1004" Status="Cancelled"/>
  </TaskCancelResponse>
</WWKS>
```

Library

See *TaskCancelResponse*.

6.11 Configurationchecking

Lead elements

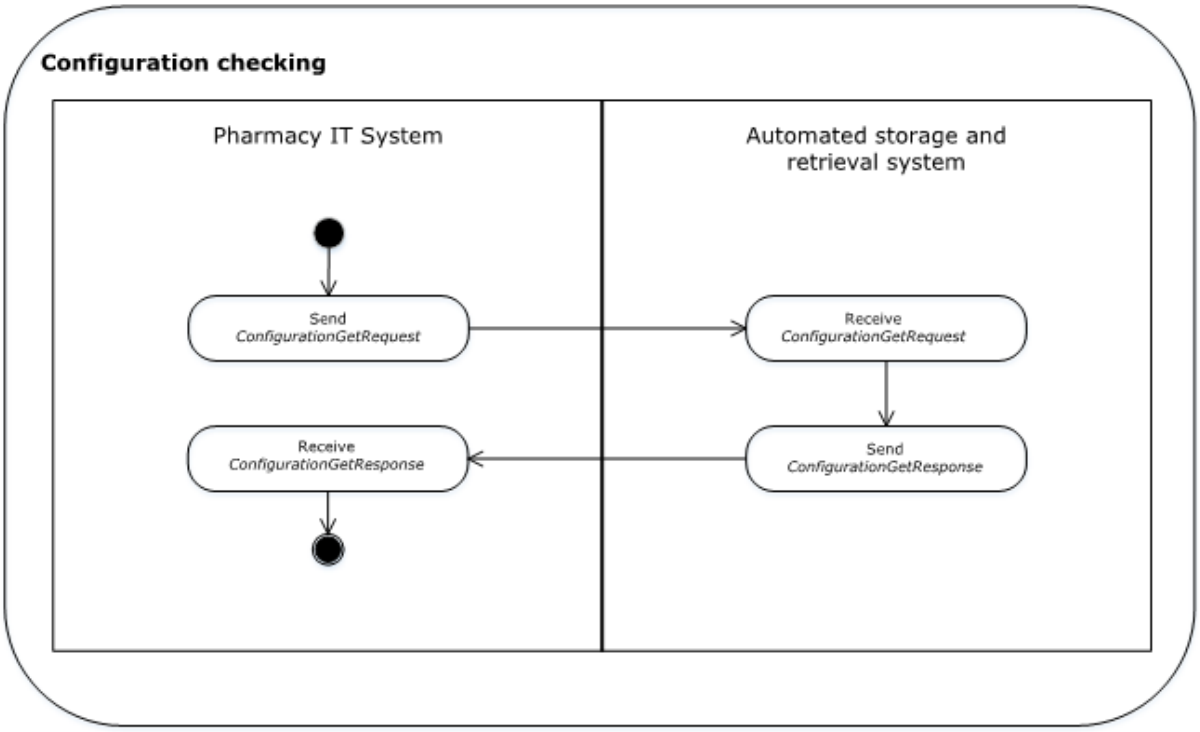
ConfigurationGetRequest

ConfigurationGetResponse

Usage

When the pharmacy IT system wants to check the current configuration of the automated storage and retrieval system (e.g. for backup or diagnostic purposes), it sends the *ConfigurationGetRequest.* The automated storage and retrieval system will respond with a *ConfigurationGetResponse*. Content and structure of the returned configuration data depend on the manufacturer of the automated storage and retrieval system.

Sequence



6.11.1 ConfigurationGetRequest

Structure

```
<WWKS>
  <ConfigurationGetRequest/>
</WWKS>
```

Elements

Element	M/O	Data type	Description
ConfigurationGetRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the configuration checking process. This ID is returned in the *ConfigurationGetResponse.*
Source	M	Integer 32-bit >0	ID of the system sending the *ConfigurationGetRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *ConfigurationGetRequest*

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <ConfigurationGetRequest Id="3335" Source="100" Destination="999"/>
</WWKS>
```

Library

The current configuration of the automated storage and retrieval system can be checked anytime via the `Configuration` feature:

```
string configuration = storageSystem.Configuration;
```

When checking the configuration, the library sends a *ConfigurationGetRequest*, waits for the corresponding *ConfigurationGetResponse*, and evaluates it.

6.11.2 ConfigurationGetResponse

Structure

```
<WWKS>
  <ConfigurationGetResponse>
    <Configuration/>
  </ConfigurationGetResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
ConfigurationGetResponse	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the configuration checking process. This ID was sent in the *ConfigurationGetRequest.*
Source	M	Integer 32-bit >0	ID of the system sending the *ConfigurationGetResponse.*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *ConfigurationGetResponse.*

Element	M/O	Data type	Description
Configuration	M	Tag	Manufacturer-specific configuration of the automated storage and retrieval system

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <ConfigurationGetResponse Id="3335" Source="999" Destination="100">
    <Configuration>
      <![CDATA[
        <ConfigurationValue Name="abc" Value="dddd"/>
        <ConfigurationValue Name="ccc" Value="ffff"/>
        <ConfigurationValue Name="ddd" Value="dfdf"/>
      ]]>
    </Configuration>
  </ConfigurationGetResponse>
</WWKS>
```

Library

See *ConfigurationGetRequest*.

6.12 Stock location checking

Lead elements

StockLocationInfoRequest

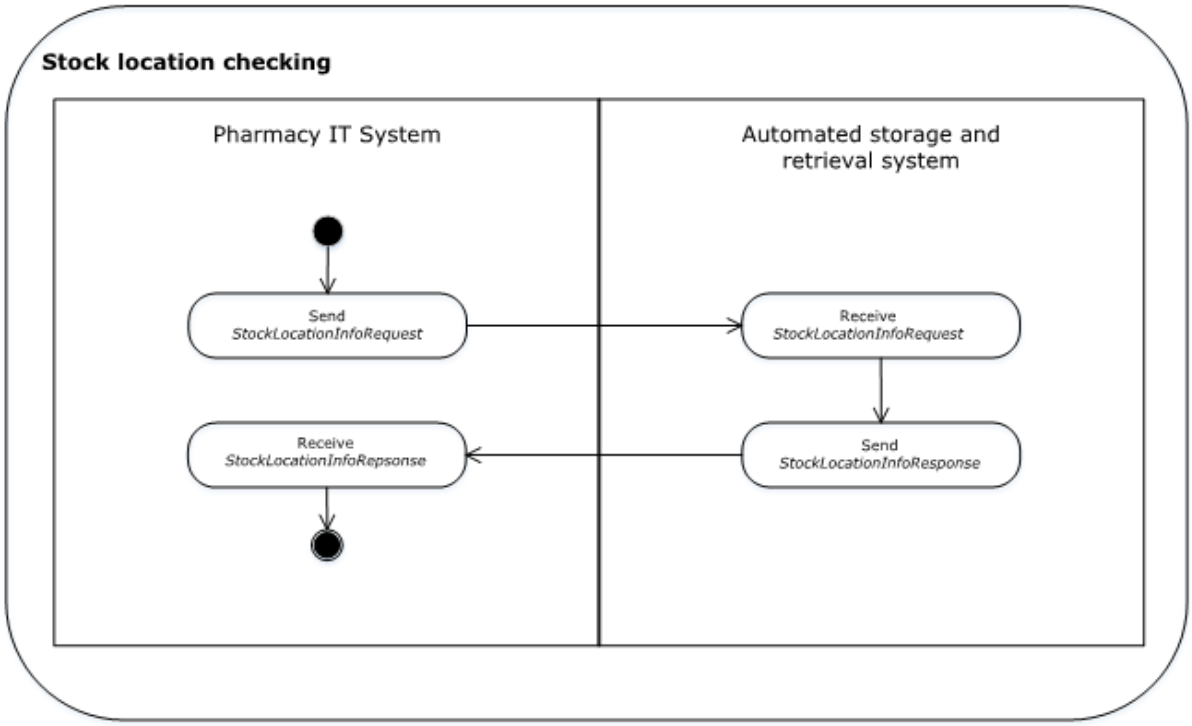
StockLocationInfoResponse

Usage

When the pharmacy IT system wants to check the currently configured stock locations of the automated storage and retrieval system, it sends the *StockLocationInfoRequest*.

The automated storage and retrieval system will respond with a *StockLocationInfoResponse*. These messages are only used when an automated storage and retrieval system is divided into several virtual stock locations.

Sequence



6.12.1 StockLocationInfoRequest

Structure

```
<WWKS>
  <StockLocationInfoRequest/>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockLocationInfoRequest	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock location request process. This ID is returned in the *StockLocationInfoResponse*.
Source	M	Integer 32-bit >0	ID of the system sending the *StockLocationInfoRequest*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StockLocationInfoRequest*

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockLocationInfoRequest Id="3335" Source="100" Destination="999"/>
</WWKS>
```

Library

The current configuration of the automated storage and retrieval system can be checked anytime via the `StockLocations` feature:

```
var stockLocations = storageSystem.StockLocations;
```

When checking the configuration, the library sends a *StockLocationInfoRequest*, waits for the corresponding *StockLocationInfoResponse*, and evaluates it.

6.12.2 StockLocationInfoResponse

Structure

```
<WWKS>
  <StockLocationInfoResponse>
    <StockLocation/>
  </StockLocationInfoResponse>
</WWKS>
```

Elements

Element	M/O	Data type	Description
StockLocationInfo Response	M	Tag	Message type
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	ID of the stock location request process. This ID was returned in the *StockLocationInfoRequest*.
Source	M	Integer 32-bit >0	ID of the system sending the *StockLocationInfoResponse.*
Destination	M	Integer 32-bit >0	ID of the system intended to receive the *StockLocationInfoResponse.*

Element	M/O	Data type	Description
StockLocation	M	Tag	Stock location information follows. This element may occur multiply.
Attributes	**M/O**	**Data type**	**Description and Values**
Id	M	String	Identification of the stock location
Description	O	String	Optional description of the stock location

Example

```
<WWKS Version="2.0" TimeStamp="2013-04-16T11:14:00Z">
  <StockLocationInfoResponse Id="3335" Source="999" Destination="100">
    <StockLocation Id="463563" Description="Narcotics" />
    <StockLocation Id="674638" Description="SmartDrugs" />
  </StockLocationInfoResponse>
</WWKS>
```

Library

See *StockLocationInfoRequest*.

7 Appendix

Further documents

More detailed information can be found in the following documents:

- *Rowa system architecture – overview*
  A detailed depiction of the Rowa system architecture in terms of network communications and security