



WWKS 2 Extension

Virtual Level Item Identification

Version 1.0

Contents

2	Introduction.....	4
2.1	Scope.....	4
2.2	Notation	4
3	Item identification levels	4
3.1	Standard WWKS2	4
3.2	Virtual Level extension.....	4
3.3	Usage sample: UK AMPP and VMP(P).....	4
4	Interface extension	6
4.1	Stock enquiry and stock change.....	6
4.1.1	StockInfoRequest.....	7
4.1.2	StockInfoResponse	10
4.2	Stock Input	14
4.2.1	InputRequest	15
4.2.2	InputResponse	20
4.3	Stock output.....	25
4.3.1	OutputRequest	27
4.3.2	OutputResponse	32
4.3.3	OutputMessage	37
4.3.4	InputMessage	46

Definition of terms

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

Abbreviation	Definition
PIS	Pharmacy IT system
ASRS	Automated storage and retrieval system. 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

2 Introduction

2.1 Scope

WWKS2 is a standard protocol to integrate automated storage and retrieval systems (ASRS) into the workflows managed by a pharmacy IT system (PIS).

This document defines an extension of the WWKS2 protocol that adds a second level of identification to the items stored in the ASRS – the “virtual level”.

For the default scope of WWKS2, please refer to the WWKS2 reference manual.

2.2 Notation

Changes and enhancements to the original specification are marked with **orange text color**.

3 Item identification levels

3.1 Standard WWKS2

Standard WWKS2 allows the PIS to map 1 or multiple barcodes found on packages to an article id that is used to manage the stock in the ASRS.

Whenever a pack is about to be loaded into the ASRS, it sends an InputRequest with the content of scanned barcode to the PIS. If the PIS accepts this item, it answers with an InputResponse defining the article id and name.

To request dispensing of an item stored in the ASRS, the PIS has to send an OutputRequest with the article id defined in the InputResponse.

3.2 Virtual Level extension

This WWKS2 extension allows the PIS to define a second identification number for the pack, the virtual id. When requesting dispensing, the PIS may either use an article id or a virtual id.

The interface does not enforce that virtual id and article id have a hierarchical relation to each other, though the PIS may use it in this way. It is even possible to have different virtual ids for packs of the same article id and vice versa.

3.3 Usage sample: UK AMPP and VMP(P)

UK National Health Service (NHS) uses a national dictionary database of medicines and medical devices (dm+d). This dictionary defines five levels of medicinal product identification.

WWKS 2 Extension – Virtual Level Item Identification

dm+d level		Identification Level	Example
Virtual Therapeutic Moiety	VTM	Active ingredient	Amoxicillin
Virtual Medicinal Product	VMP	Active ingredient, strength, dosage form	Amoxicillin 500mg capsules
Virtual Medicinal Product Pack	VMPP	Active ingredient, strength, dosage form, pack size	Amoxicillin 500mg capsules 21 capsules
Actual Medicinal Product	AMP	Manufacturer/Brand, strength, dosage form	Amoxil 500mg capsules (GlaxoSmithKline)
Actual Medicinal Product Pack	AMPP	Manufacturer/Brand, strength, dosage form, pack size	Amoxil 500mg capsules (GlaxoSmithKline) 21 capsules

Stock in a ASRS is usually managed on AMPP level using the article id, while prescriptions are managed on VMP or VMPP level. If the virtual id is used to store a VMP(P) code, the ASRS will choose the manufacturers used to fulfill the request.

4 Interface extension

4.1 Stock enquiry and stock change

Lead elements

StockInfoRequest

StockInfoResponse

StockInfoMessage

Usage

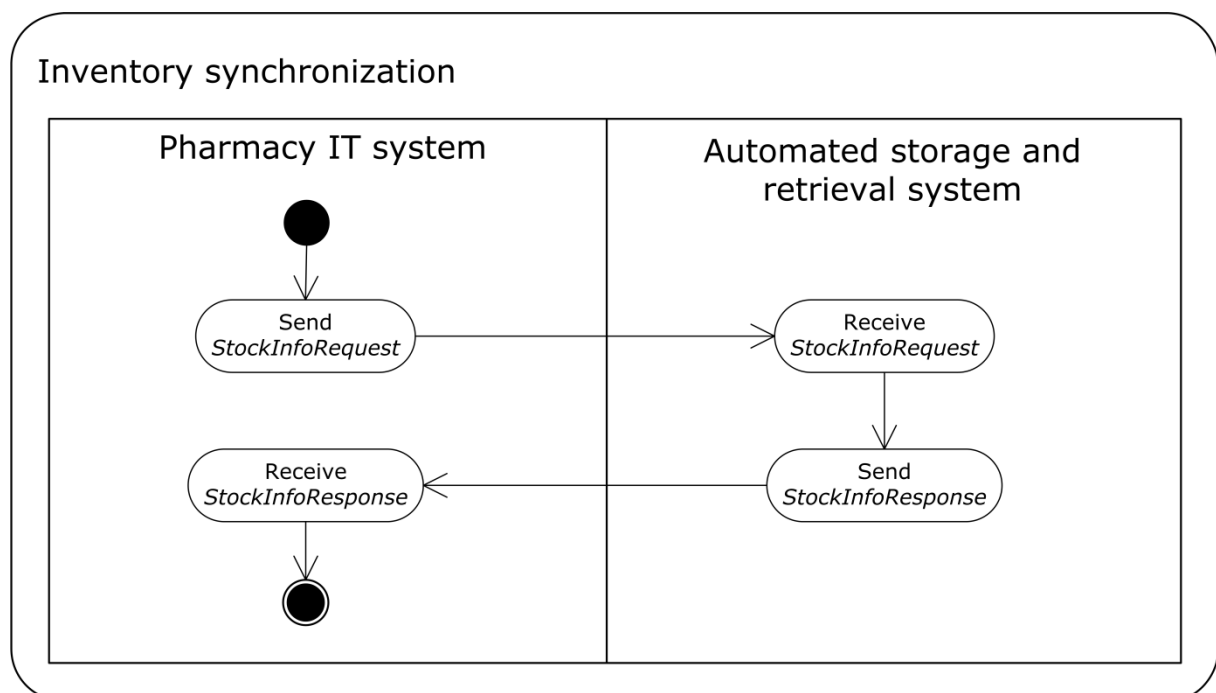
To poll the current stock level in the automated storage and retrieval system, the pharmacy IT system can send the *StockInfoRequest*. The automated storage and retrieval system will respond with *StockInfoResponse*.

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).

The ASRS accepts both article id and virtual id in the *ArticleId* search criteria attribute. It will first search for packs that have the given value stored in its article id. If no pack is found, it will continue to search for packs that have the given value as virtual id.

Sequence



4.1.1 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 <i>StockInfoResponse</i> .
Source	M	Integer 32-bit >0	ID of the system sending the <i>StockInfoRequest</i> .
Destination	M	Integer 32-bit >0	ID of the system to which the <i>StockInfoRequest</i> is sent.
IncludePacks	O	Boolean	This flag specifies whether details of the existing packs are to be returned. Possible values: "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 values: "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 either the article id or the virtual id assigned by the pharmacy IT system when placing the article into stock in the <i>StockInputResponse</i> .
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.
SerialNumber	O	String	By setting this filter, only packs with the specified serial number 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"/>
  </StockInfoRequest>
</WWKS>
```



```
<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");
```

4.1.2 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 <i>StockInfoRequest</i> .
Source	M	Integer 32-bit >0	ID of the system sending the <i>StockInfoResponse</i> .
Destination	M	Integer 32-bit >0	ID of the system intended to receive the <i>StockInfoResponse</i> .

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 <i>InputResponse</i> .
VirtualId	O	String	Virtual ID of the article. This ID corresponds to the one assigned by the pharmacy IT system when placing the article into stock in the <i>InputResponse</i> .

WWKS 2 Extension – Virtual Level Item Identification

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	M	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 <i>InputResponse</i> during stock input.
ExternalId	O	String	External ID, an additional ID of the pack as sent in the <i>InputResponse</i> on stock input.
ExternalId	O	String	Serial number of the pack.
ExpiryDate	O	String	Expiration 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

WWKS 2 Extension – Virtual Level Item Identification

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 values: "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 values: "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="2019-01-18T10:55:22Z">

```
<StockInfoResponse Id="240" Source="999" Destination="100">
  <Article Id="23226921000171118" Name=" Galpharm Hayfever+ Allergy Relief 10mg tabs"
    DosageForm="" PackagingUnit="14" VirtualId="22181451000171112" Quantity="1"
    MaxSubItemQuantity="14" />
  <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*.

4.2 Stock Input

The Stock Input workflow of WWKS2 is not changed. *InputResponse* and *InputMessage* define additional attributes *VirtualId* and *VirtualName* for setting the virtual level identification details.

Lead elements

InputRequest

InputResponse

InputMessage

Usage

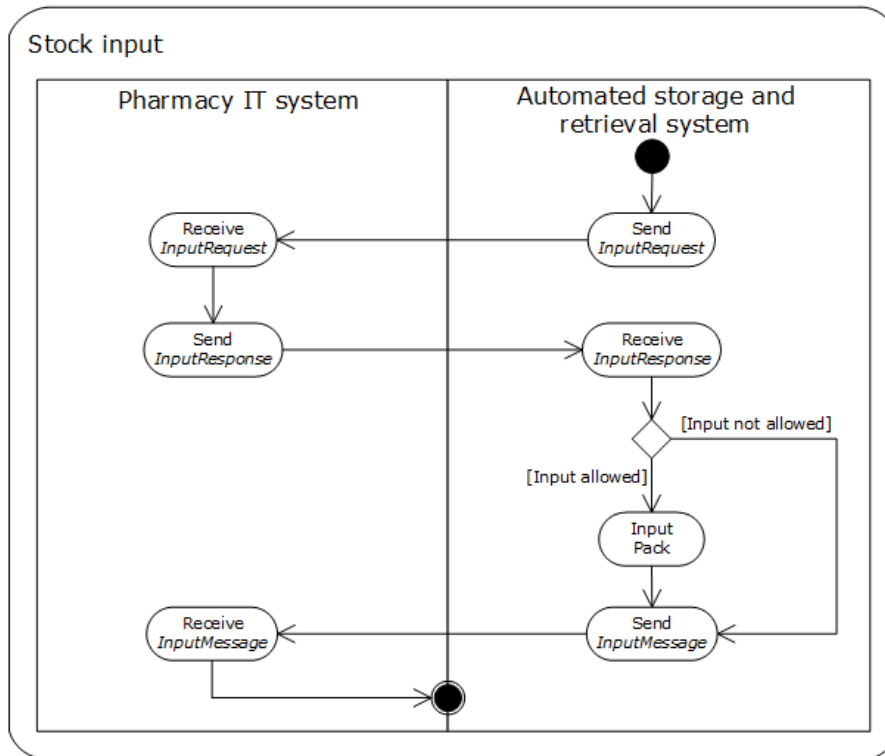
When a user or an automatic stock input system is wanting to store a pack in the automated storage and retrieval system, the *InputRequest* is sent from the automated storage and retrieval system to the pharmacy IT system. The pharmacy IT system must respond to the stock input request with the *InputResponse*.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



4.2.1 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 <i>InputResponse</i> and is used in the associated <i>InputMessage</i> .

WWKS 2 Extension – Virtual Level Item Identification

Source	M	Integer 32-bit >0	ID of the system sending the <i>InputRequest</i>
Destination	M	Integer 32-bit >0	ID of the system receiving this message
IsNewDelivery	O	Boolean	<p>This flag identifies the stock input request as part of a stock delivery.</p> <p>"True" means that a stock delivery is taking place. The attribute <i>DeliveryNumber</i> in the element <i>Pack</i> should then contain the stock delivery number. Blank stock delivery numbers (e.g. "") are possible in some customer-specific scenarios.</p> <p>"False" means that the stock input relates to a stock return.</p> <p>The default value is "False".</p>
SetPickingIndicator	O	Boolean	<p>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".</p> <p>This case occurs primarily on initial stocking of the automated storage and retrieval system.</p> <p>The default value is "False".</p>

Element	M/O	Data type	Description
Article	M	Tag	Article information follows.
Index	O	String	Article identification code extracted from the barcode of the pack.

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.

WWKS 2 Extension – Virtual Level Item Identification

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 <i>IsNewDelivery</i> of the element <i>InputRequest</i> 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.
SerialNumber	O	String	Serial number of the pack. This attribute is used when the barcode of the pack contains or the user has entered a serial number.
ExpiryDate	O	String	Expiration date of the pack in format YYYY-MM-DD. This attribute is used when the picker has entered an expiration 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 (linear barcode scanned)

```
<WWKS Version="2.0" TimeStamp="2019-01-18T10:02:18Z">
  <InputRequest Id="112240" Source="999" Destination="100" IsNewDelivery="True"
    SetPickingIndicator="False">
    <Article Id="5017353503558">
      <Pack Index="0" ScanCode="5017353503558" DeliveryNumber="ABC123" BatchNumber=""
        ExternalId="" SerialNumber="" SubItemQuantity="0" StockLocationId=""
        MachineLocation="999" />
    </Article>
  </InputRequest>
</WWKS>
```

Example (data matrix code scanned)

```
<WWKS Version="2.0" TimeStamp="2019-12-09T11:36:15Z">
  <InputRequest Id="100048" Source="999" Destination="100" IsNewDelivery="False"
    SetPickingIndicator="False">
    <Article Id="18901079001402">
      <Pack Index="0" ScanCode="011890107900140210AKS028004\x1D17200100218S1W4D2ATVVS"
        BatchNumber="AKS028004" ExternalId="8S1W4D2ATVVS"
        SerialNumber="8S1W4D2ATVVS" ExpiryDate="2020-01-01" SubItemQuantity="0"
        StockLocationId="" MachineLocation="999" />
    </Article>
  </InputRequest>
</WWKS>
```

Library

To handle a stock input process, the user must register for the following events:

PackInputRequested or PackInputRequestedExtended

PackStored or PackStoredExtended

PackInputFinished or PackInputFinishedExtended

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

```
storageSystem.PackInputRequestedExtended +=
  StorageSystem_PackInputRequestedExtended;
storageSystem.PackStoredExtended +=
  StorageSystem_PackStoredExtended;
```

When one or more packs is awaiting stock input, the automated storage and retrieval system sends an *InputRequest*. This triggers the *PackInputRequested* or *PackInputRequestedExtended* event depending on which you registered for. 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 expiration date or batch number.

WWKS 2 Extension – Virtual Level Item Identification

```
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;

        string virtualId = "V12345";

        // PUT DECISION LOGIC HERE

        // define article information for this pack
        pack.SetArticleInformation(articleId,
                                   "My Article Name",
                                   "Dosage Form",
                                   "Packaging Unit",
                                   14, // MaxSubItemQuantity
                                   virtualId,
                                   "My Virtual Name" );

        // allow pack input
        pack.SetHandling(InputHandling.Allowed);
    }
    // send InputResponse
    request.Finish();
}

void StorageSystem_PackInputRequestedExtended(object sender,
PackInputRequestedEventArgs eventArgs)
{
    // for now just accept every input request
    foreach (var pack in eventArgs.Request.Packs)
    {
        Console.WriteLine("Pack '{0}' requested input via request '{1}'
from source '{2}'.", pack.ScanCode, eventArgs.Request.Id,
eventArgs.Request.Source);

        var articleCode = pack.ScanCode;
        if (articleCode.Length < 8)
            articleCode = articleCode.PadLeft(8, '0');

        var virtualCode = 'V' + articleCode;

        // set dummy article information
        pack.SetArticleInformation(articleCode,
                                   string.Format("My Article {0}",
                                   articleCode),
                                   "Dosage Form",
                                   "Packaging Unit",
                                   0,
                                   string.Format("Virtual-{0}",
                                   virtualCode),
                                   string.Format("My Virtual Article {0}",
                                   virtualCode));

        // allow pack input
        pack.SetHandling(InputHandling.Allowed);
    }

    eventArgs.Request.Finish();
}
```

where PackInputRequestedEventArgs has the following structure:

```

/// <summary>
/// The PackInputRequested event arguments.
/// </summary>
public class PackInputRequestedEventArgs : EventArgs
{
    /// <summary>
    /// Initializes a new instance of the <see cref="PackInputRequestedEventArgs"/>
class.
    /// </summary>
    /// <param name="request"></param>
    public PackInputRequestedEventArgs(IInputRequest request)
    {
        Request = request;
    }

    /// <summary>
    /// Gets the details about the requested pack input.
    /// </summary>
    public IInputRequest Request { get; }
}

```

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 expiration 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.

4.2.2 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

WWKS 2 Extension – Virtual Level Item Identification

Id	M	String	ID of the stock input process. This ID was sent in the <i>InputRequest</i> .
Source	M	Integer 32-bit >0	ID of the system sending the <i>InputResponse</i>
Destination	M	Integer 32-bit >0	Corresponds to the value of Source in the associated <i>InputRequest</i> .
IsNewDelivery	O	Boolean	This flag identifies the stock input request as part of a stock delivery. The same value as in the associated <i>InputRequest</i> 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).
VirtualId	O	String	Virtual level article id used as alternative identification of the article. Both article id and virtual id may be used to request dispensing of this pack.
VirtualName	O	String	Name of virtual level article. The automated storage and retrieval system displays the name in its own inventory view (GUI).
DosageForm	O	String	Dosage form of the article.

WWKS 2 Extension – Virtual Level Item Identification

			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).
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 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 <i>InputResponse</i> covers multiple packs. The index number corresponds to the one in the <i>InputRequest</i> .
DeliveryNumber	O	String	Stock delivery number specified in the <i>InputRequest</i>
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).
SerialNumber	O	String	Serial number to be saved for this pack.
ExpiryDate	O	String	Expiration date in format YYYY-MM-DD

WWKS 2 Extension – Virtual Level Item Identification

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 <i>InputRequest</i> 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	<p>Handling instructions for the pack. Possible values:</p> <p>"Allowed" if stock input is allowed</p> <p>"AllowedForFridge" if the article needs to be refrigerated</p> <p>"Rejected" if stock input is not allowed</p> <p>"RejectedNoExpiryDate" if stock input is not allowed because the expiration date was not specified</p> <p>"RejectedNoPickingIndicator" if stock input is not allowed because the article is not declared as capable of automated handling</p> <p>"RejectedNoBatchNumber" if stock input is not allowed because the batch number was not specified in the automated storage and retrieval system</p> <p>"RejectedNoStockLocation" if stock input is not allowed because the stock location ID was not specified in the automated storage and retrieval system</p> <p>"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.</p> <p>"RejectedNoSerialNumber" if stock input is not allowed because the serial number was not specified in the automated storage and retrieval system</p>

WWKS 2 Extension – Virtual Level Item Identification

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="2019-01-18T10:02:17Z">
  <InputResponse Id="112240" Source="100" Destination="999" IsNewDelivery="True">
    <Article Id="23226921000171118" Name="Galpharm Hayfever+ Allergy Relief 10mg tabs"
      DosageForm="" PackagingUnit="14" RequiresFridge="False"
      VirtualId="22181451000171112"
      VirtualName="Galpharm Hayfever+ Allergy Relief 10mg tabs" MaxSubItemQuantity="14">
      <Pack Index="0" ScanCode="5017353503558" DeliveryNumber="" BatchNumber="" ExternalId=""
        SerialNumber="" SubItemQuantity="14" StockLocationId="" MachineLocation="999">
        <Handling 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="Barcode unknown." />
      </Pack>
    </Article>
  </InputResponse>
</WWKS>
```

Library

See *InputRequest*.

4.3 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.

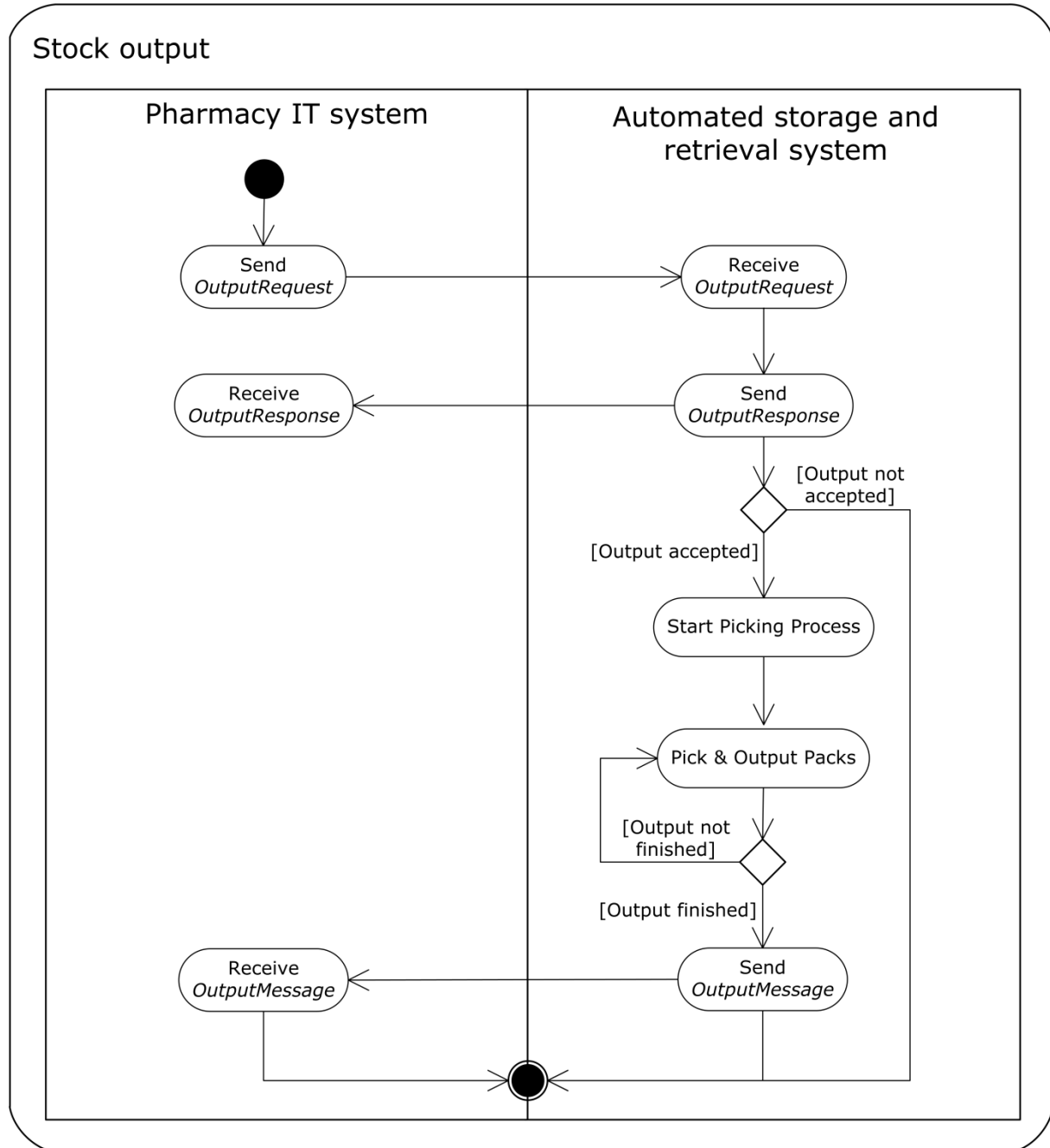
Some automated storage and retrieval systems offer the facility to request packs directly on the automated storage and retrieval system without going through the pharmacy IT system. This is done by an operative on the automated storage and retrieval system's user interface. This procedure requires a 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 does not therefore 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*.

The ASRS accepts both article id and virtual id in the ArticleId search criteria attribute. It will first search for packs that have the given value stored in its article id. If no pack is found, it will continue to search for packs that have the given value as virtual id.

Sequence



4.3.1 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 <i>OutputResponse</i> and is also used in the associated <i>OutputMessage</i> .
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

WWKS 2 Extension – Virtual Level Item Identification

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 either the article id or the virtual Id in the <i>InputResponse</i> . The ASRS will first try to match to an article id in stock. Only if no pack is found, it will try to match to a virtual id.
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 <i>Quantity</i> attribute is ignored. It should nevertheless be set to "0".
MinimumExpiryDate	O	String	Filter for packs having the specified expiration 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.

WWKS 2 Extension – Virtual Level Item Identification

ExternalId	O	String	Filter for packs having the specified external ID
SerialNumber	O	String	Filter for packs having the specified serial number.
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. There are two versions of the method: one taking the order number as `int` and another one taking the order number as

`string`).

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` (as `int`) 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();
```

In the following example, an output order with the number `1234` (as `string`) 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();
```

4.3.2 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 <i>OutputRequest</i> .
Source	M	Integer 32-bit >0	ID of the system sending the <i>OutputResponse</i>
Destination	M	Integer 32-bit >0	ID of the system receiving the <i>OutputResponse</i>
BoxNumber	O	String	Number of the box specified in the <i>OutputRequest</i> . 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

WWKS 2 Extension – Virtual Level Item Identification

Priority	O	String	Priority of this stock output process. Here the same value is to be seen as in the <i>OutputRequest</i> .
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 <i>OutputRequest</i> .
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 <i>OutputRequest</i> .
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 <i>OutputRequest</i> .
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 <i>StockInputResponse</i> .
Quantity	M	Integer 32-bit ≥0	Number of full packs to output. Here the same value is to be seen as in the <i>OutputRequest</i> .
SubItemQuantity	O	String	Number of units (e.g. tablets, ampules) to be outputted. If this attribute is used, the <i>Quantity</i> attribute is ignored. It should nevertheless be set to "0". Here the same value is to be seen as in the <i>OutputRequest</i> .
MinimumExpiryDate	O	Integer 32-bit ≥0	Filter for packs having the specified expiration date as a minimum. Format YYYY-MM-DD. Here the same value is to be seen as in the <i>OutputRequest</i> .
BatchNumber	O	String	Filter for packs having the specified batch number. Here the same value is to be seen as in the <i>OutputRequest</i> .

WWKS 2 Extension – Virtual Level Item Identification

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 <i>OutputRequest</i> .
ExternalId	O	String	Filter for packs having the specified external ID. Here the same value is to be seen as in the <i>OutputRequest</i> .
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 <i>OutputRequest</i> .
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 <i>OutputRequest</i> .
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 <i>OutputRequest</i> .

Element	M/O	Data type	Description
Label	O		Label information follows. This element may occur multiply. The data corresponds to that in the <i>OutputRequest</i> .
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 <i>OutputRequest</i> .

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 <i>OutputRequest</i> .
---------	---	--	--

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>NIFEDIPIIN 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>
```

```

        </labeldosageline>
    </dosagelines>
  ]]>
</Content>
</Label>
</Criteria>
</OutputResponse>
</WWKS>

```

Library

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

- Option 1 (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());
}

```

- Option 2: 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);
}

```

4.3.3 OutputMessage

Structure

```
<WWKS>
  <OutputMessage>
    <Details/>
    <Article>
      <Pack/>
    </Article>
  </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 <i>OutputRequest</i> .
Source	M	Integer 32-bit >0	ID of the system sending the <i>OutputMessage</i>
Destination	M	Integer 32-bit >0	ID of the system receiving the <i>OutputMessage</i>

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 <i>OutputRequest</i> .
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 <i>OutputRequest</i> .

WWKS 2 Extension – Virtual Level Item Identification

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 <i>OutputRequest</i> .
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 "BoxReleased" if a box was successfully released

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
VirtualId	O	String	Virtual 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

WWKS 2 Extension – Virtual Level Item Identification

ExpiryDate	O	String	Expiration 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 values: "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.

WWKS 2 Extension – Virtual Level Item Identification

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 <i>OutputRequest</i> or automatically determined during filling.

Example of a successfully completed stock output

```
<WWKS Version="2.0" TimeStamp="2019-01-18T12:43:15Z">
  <OutputMessage Id="4142" Source="999" Destination="100">
    <Details Priority="Normal" OutputDestination="0" Status="Completed" />
    <Article Id="23226921000171118" VirtualId="22181451000171112" >
      <Pack Id="8589935727" ScanCode="5017353503558" DeliveryNumber="" BatchNumber=""
        ExternalId="" SerialNumber="" ExpiryDate="2020-07-18" StockInDate="2019-01-18"
        SubItemQuantity="14" Depth="88" Width="73" Height="20" Shape="Cuboid"
        State="Available" IsInFridge="False" StockLocationId=""
        MachineLocation="999"/>
    </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="Comple
```

```

ed"/>
    <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);
    }
}

```

As soon as a box has been released, the status and a detailed list of outputted packs for the released box can be polled:

```

void OutputProcess_BoxReleased(object sender, EventArgs e)
{
    var output = sender as IOutputProcess;

    Console.WriteLine("Output process '{0}' finished with result '{1}'.",
        output.OrderNumber, output.State.ToString());

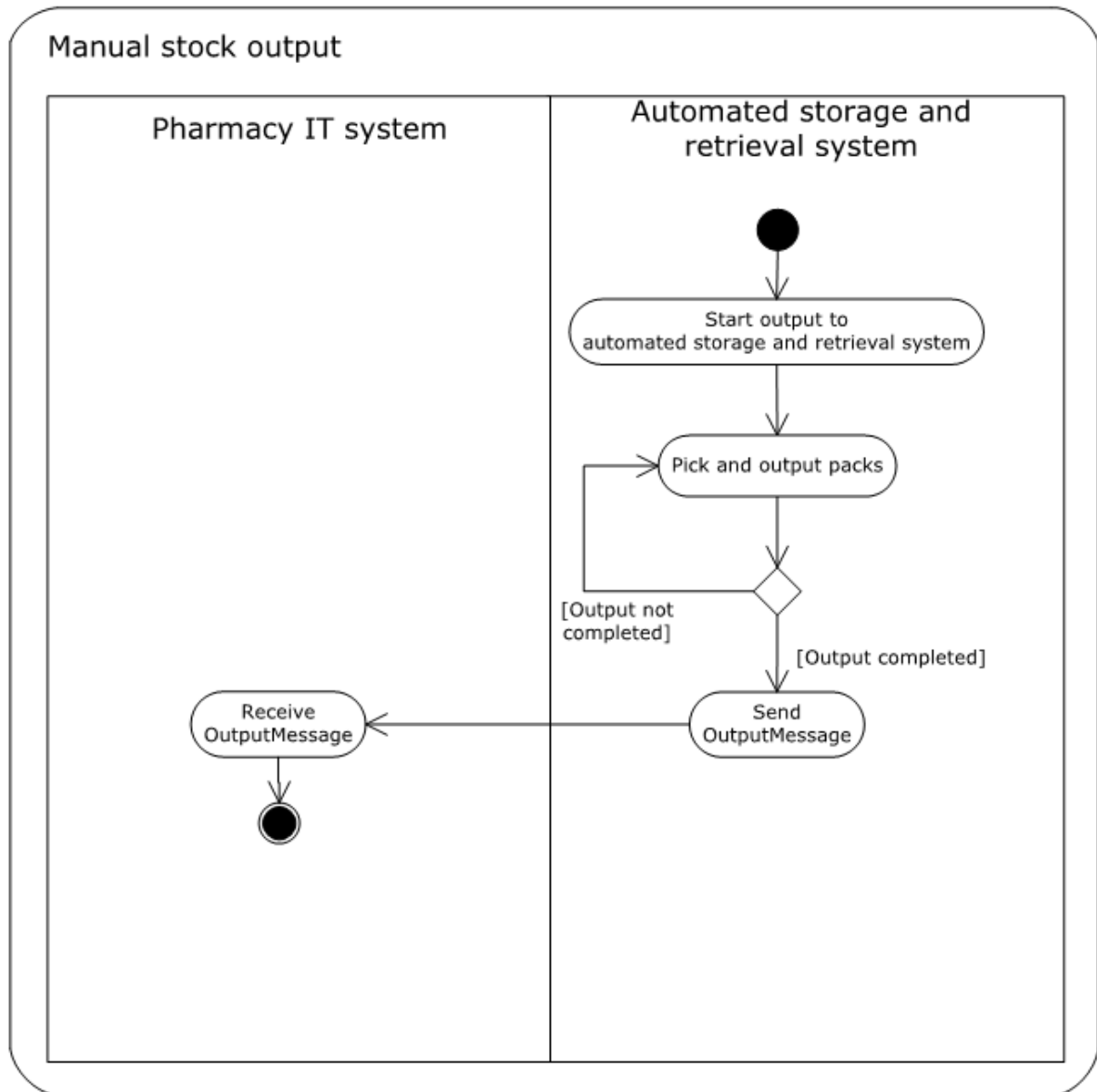
    foreach (var pack in output.Packs)
    {
        Console.WriteLine("Pack '{0}' with ScanCode '{1}' and StockInDate
            '{2}' was dispensed by output process {3}.", pack.Id, pack.ScanCode,
            pack.StockInDate.ToShortDateString(), output.OrderNumber);
    }
}

```

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 one of the following events `PackDispensed` or `PackDispensedExtended`. These events are called whenever a pack has been outputted without an output order.

```
storageSystem.PackDispensed += StorageSystem_PackDispensed;
```

```
storageSystem.PackDispensedExtended +=
StorageSystem_PackDispensedExtended;
```

```
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);
        }
    }
}

void StorageSystem_PackDispensedExtended(object sender,
PackDispensedEventArgs eventArgs)
{
    foreach (var article in eventArgs.Articles)
    {
        foreach (var pack in article.Packs)
        {
            Console.WriteLine("Pack '{0}' for article '{1}' from
order '{2}' has been dispensed.", pack.Id, article.Id,
```

```
eventArgs.OrderNumber);
    }
}
}
```

where `PackDispensedEventArgs` has the following structure:

```
/// <summary>
/// The PackDispensed event arguments.
/// </summary>
public class PackDispensedEventArgs : EventArgs
{
    /// <summary>
    /// Initializes a new instance of the <see
    cref="PackDispensedEventArgs"/> class.
    /// </summary>
    /// <param name="articles">The list of articles with the
    packs that were dispensed.</param>
    /// <param name="orderNumber">The order number.</param>
    public PackDispensedEventArgs(IArticle[] articles, string
    orderNumber)
    {
        Articles = articles;
        OrderNUmber = orderNumber;
    }

    /// <summary>
    /// Gets the information of the articles and the packs that
    were dispensed.
    /// </summary>
    public IArticle[] Articles { get; }

    /// <summary>
    /// Gets the order number.
    /// </summary>
    public string OrderNUmber { get; }
}
```

4.3.4 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 <i>InputRequest</i> .
Source	M	Integer 32-bit >0	ID of the system sending the <i>InputMessage</i>
Destination	M	Integer 32-bit >0	ID of the system intended to receive the <i>InputMessage</i>
IsNewDelivery	O	Boolean	This flag identifies the stock input as part of a stock delivery. The same value as in the associated <i>InputRequest</i> 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 <i>InputResponse</i>

WWKS 2 Extension – Virtual Level Item Identification

Name	O	String	Name of the article sent in the <i>InputResponse</i>
VirtualId	O	String	The virtual ID sent in the <i>InputResponse</i>
VirtualName	O	String	The virtual name sent in the <i>InputResponse</i>
DosageForm	O	String	The dosage form sent in the <i>InputResponse</i>
PackagingUnit	O	String	The packaging unit sent in the <i>InputResponse</i>
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 <i>InputResponse</i> .

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 <i>InputMessage</i> covers multiple packs. The index number corresponds to the one in the <i>InputRequest</i> .
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 <i>InputRequest</i>
BatchNumber	O	String	Batch number saved for this pack
ExternalId	O	String	External ID specified in the <i>InputResponse</i> and saved for this pack
SerialNumber	O	String	Serial number saved for this pack.
ExpiryDate	O	String	Expiration date in format YYYY-MM-DD

WWKS 2 Extension – Virtual Level Item Identification

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.
IsInFridge	O	Boolean	Flag indicating whether the pack is stored in a refrigeration unit ("True"). The default value is "False".
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="2019-01-18T10:02:22Z">
  <InputMessage Id="112240" Source="999" Destination="100" IsNewDelivery="True">
    <Article Id="23226921000171118" Name="Galpharm Hayfever+ Allergy Relief 10mg tabs"
      DosageForm="" PackagingUnit="14" VirtualId="22181451000171112"
      VirtualName="Galpharm Hayfever+ Allergy Relief 10mg tabs" MaxSubItemQuantity="14">
      <Pack Id="8589935727" Index="0" ScanCode="5017353503558" DeliveryNumber=""
        BatchNumber="" ExternalId="" SerialNumber="" ExpiryDate="2020-07-18"
        StockInDate="2019-01-18" SubItemQuantity="14" Depth="88" Width="73" Height="20"
        Shape="Cuboid" State="Available" IsInFridge="False" StockLocationId=""
        MachineLocation="999">
        <Handling Input="Completed" Text="" />
      </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">
      </Pack>
    </Article>
  </InputMessage>
</WWKS>
```

```

        <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* or *PackStoredExtended* event depending on which you registered for. 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);
        }
    }
}

void StorageSystem_PackStoredExtended(object sender, PackStoredEventArgs eventArgs)
{
    foreach (var article in eventArgs.Articles)
    {
        foreach (var pack in article.Packs)
        {
            Console.WriteLine("Pack '{0}' for article '{1}' was stored.
Input was a new delivery: {2}.", pack.Id, article.Id, eventArgs.IsNewDelivery);
        }
    }
}

```

where *PackStoredEventArgs* has the following structure:

```

/// <summary>
/// The PackStored event arguments.
/// </summary>
public class PackStoredEventArgs : EventArgs
{
    /// <summary>
    /// Initializes a new instance of the <see cref="PackStoredEventArgs"/>
class.
    /// </summary>
    /// <param name="articles">The list of articles with the packs that were
stored.</param>
    /// <param name="isNewDelivery"><c>true</c> or <c>false</c> wheter the
input was a new delivery or not.</param>
    public PackStoredEventArgs(IArticle[] articles, bool isNewDelivery)
    {
        Articles = articles;
        IsNewDelivery = isNewDelivery;
    }

    /// <summary>
    /// Gets the information of the articles and the packs that were stored.
    /// </summary>

```

```

public IArticle[] Articles { get; }

/// <summary>
/// Gets <c>true</c> or <c>false</c> wheter the input was a new delivery or
not.
/// </summary>
public bool IsNewDelivery { get; }
}

```

When a connected storage system finishes an input triggers the PackInputFinished or PackInputFinishedExtended event depending on which you registered for.

```

void StorageSystem_PackInputFinished(IStorageSystem sender, int
source, string inputRequestId, InputResult result, IArticle[]
articleList)
{
    Console.WriteLine("Input request '{0}' of source {1}
finished with result '{2}'.", inputRequestId, source,
result.ToString());
}

```

```

void StorageSystem_PackInputFinishedExtended(object sender,
PackInputFinishedEventArgs eventArgs)
{
    Console.WriteLine("Input request '{0}' of source {1}
finished with result '{2}'.", eventArgs.InputRequestId,
eventArgs.Source, eventArgs.Result.ToString());
}

```

where PackInputFinishedEventArgs has the following structure:

```

/// <summary>
/// The PackInputFinished event arguments.
/// </summary>
public class PackInputFinishedEventArgs : EventArgs
{
    /// <summary>
    /// Initializes a new instance of the <see
    cref="PackInputFinishedEventArgs"/> class.
    /// </summary>
    /// <param name="sender">Object instance which raised the
    event.</param>
    /// <param name="source">The identifier of the according
    input request source that has been finished.</param>
    /// <param name="inputRequestId">The identifier of the
    according input request that has been finished.</param>
    /// <param name="result">The result of the input request
    that has been finished.</param>
    /// <param name="articles">List of articles with the packs
    that were stored.</param>
    public PackInputFinishedEventArgs(int source, int
    inputRequestId, InputResult result, IArticle[] articles)
    {
        Source = source;
    }
}

```

```

        InputRequestId = inputRequestId;
        Result = result;
        Articles = articles;
    }

    /// <summary>
    /// Gets the identifier of the according input request
source that has been finished.
    /// </summary>
    public int Source { get; }

    /// <summary>
    /// Gets the identifier of the according input request that
has been finished.
    /// </summary>
    public int InputRequestId { get; }

    /// <summary>
    /// Gets the result of the input request that has been
finished.
    /// </summary>
    public InputResult Result { get; }

    /// <summary>
    /// Gets the information of the articles and the packs that
were dispensed.
    /// </summary>
    public IArticle[] Articles { get; }
}

```