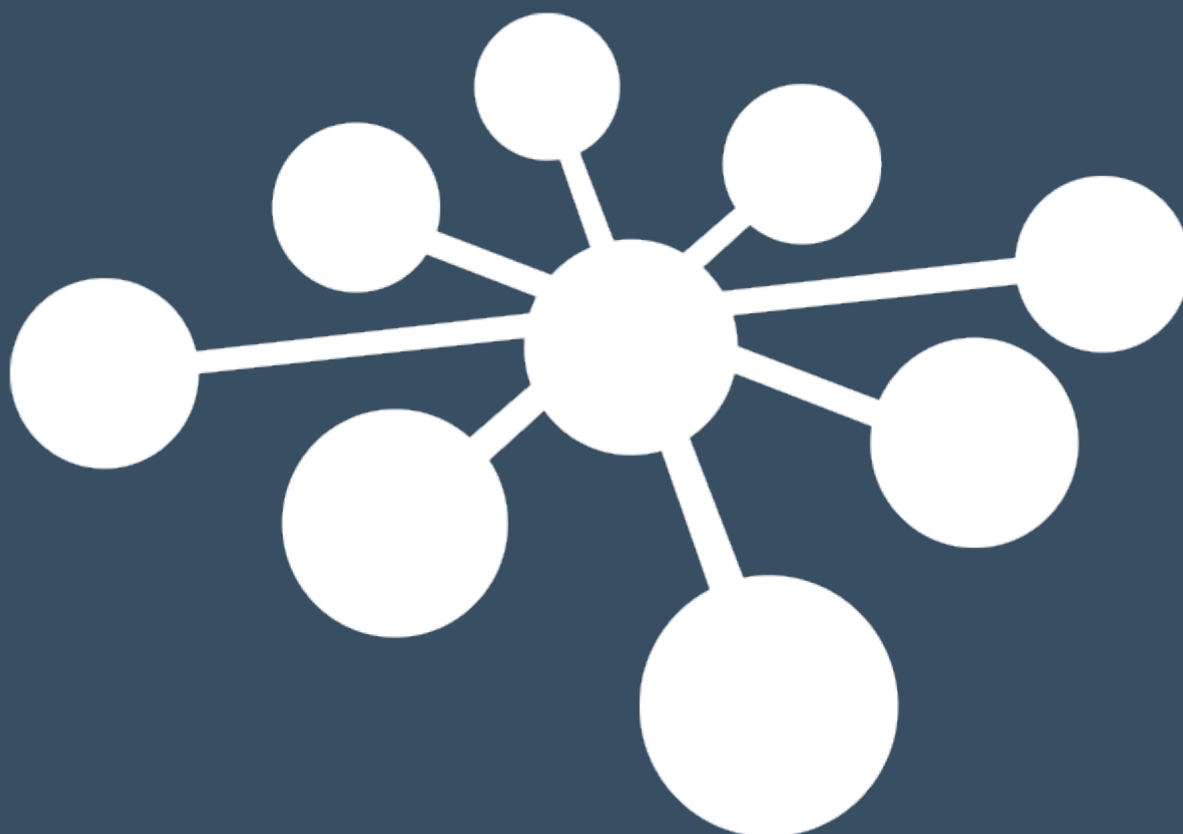


Compatibility checklist
OtoAccess[®]

Worklist HL7



Title:	Solution compatibility checklist
Project:	OtoAccess® Worklist HL7
Author:	Søren Bo Petersen, Senior Product Manager
Date issued:	July 2020

Table of Contents

1	Introduction	2
2	Intended audience	2
3	Purpose	2
4	Manuals	2
5	Licensing	2
6	Prerequisites	2
7	Default HL7 workflow	2
8	Basic solution infrastructure	3
8.1	Configuration and security	3
8.2	Searchable Category fields	4
8.3	Policies	4
8.4	Default message formats	5
9	Message configuration	5
9.1	Incoming message configuration	5
9.2	Example – AddOrder message	6
9.3	Other incoming message types	7
9.4	Order related messages	7
9.4.1	Patient only related messages	7
10	Outgoing message configuration	9
10.1	Result setup and format	10
10.2	Result delivered to file share	10
11	Message examples	10
11.1	Order messages	10
11.2	Patient messages	11
11.3	Result messages	11
11.4	Using ADT instead of ORM example	11
	Checklist	12
	Appendix 1: Available message fields	13
	Appendix 2: Messageformats.xml configuration file	14



1 Introduction

OtoAccess® Worklist HL7 provides hearing and balance clinicians with an easy to use and fast workflow solution to perform tests on supported equipment.

OtoAccess® Worklist HL7 is flexible to set up and fast to implement based on a HL7 version 2.x order flow.

2 Intended audience

This document is for technical staff with HL7 v2.x knowledge, who can determine if OtoAccess® Worklist HL7 is a compatible solution with their EMR/EHR solution and setup.

3 Purpose

This document provides a clear overview of the solution's capabilities and configuration options. This will provide the necessary information to determine if OtoAccess® Worklist HL7 is compatible with the intended use. Ultimately, this document clarifies whether you should purchase the solution or not. Ultimately, this document clarifies whether you should purchase the solution or not.

Furthermore, this document supports the Instructions for Use for OtoAccess® Worklist HL7. This will clarify configuration options for technical staff when implementing.

4 Manuals

For more comprehensive documentation, please refer to the Instructions for Use for OtoAccess® Worklist HL7. In addition, as OtoAccess® Worklist HL7 relies on OtoAccess® Database, please refer to OtoAccess® Database's Instructions for Use.

5 Licensing

OtoAccess® Worklist HL7 is a licensed software. The OtoAccess® Worklist HL7 server will require a product key and activation to function beyond the 30-day trial period.

The OtoAccess® Worklist HL7 client is not licensed and does not require activation.

All clients and servers in the setup will require OtoAccess® Database version 2.0 or newer in a client/server setup. OtoAccess® Database requires license activation on both servers and clients, please refer to the OtoAccess® Database documentation.

6 Prerequisites

HL7 v2.x order-based capability.

.Net framework 4.7.2.

OtoAccess® Database version 2.0 or newer.

7 Default HL7 workflow

The workflow is order based and the supported workflow is:

1. EMR/EHR system sends order messages to the OtoAccess® Worklist HL7 server.
 - a. Worklist HL7 server returns ACK message.
2. Clinician performs required test or tests in the OtoAccess® Worklist HL7 user interface and triggers result message (tests can also be performed in OtoAccess® Database).
3. The clinician sends the result to the EHR/EMR system via OtoAccess® Worklist HL7.
4. EMR/EHR systems return an ACK message (optional).



8 Basic solution infrastructure

OtoAccess® Worklist HL7 is an order-based solution, where the EMR/EHR system must send order messages to the OtoAccess® Worklist HL7 server containing relevant information. The solution relies on OtoAccess® Database, which is mandatory for servers *and* for clients.

The message information is divided into worklist order details and patient/measurement data. The order details are saved in an SQL CE database related to OtoAccess® Worklist HL7, whereas the patient and measurement data are saved in the SQL database related to OtoAccess® Database, see Figure 1.

The OtoAccess® Database SQL can be placed in a centralized SQL server or locally via the built-in SQL Express solution. Best practice is in a centralized SQL server. The clients must have the OtoAccess® Database and OtoAccess® Worklist HL7 clients installed and can have multiple testing solutions connected.

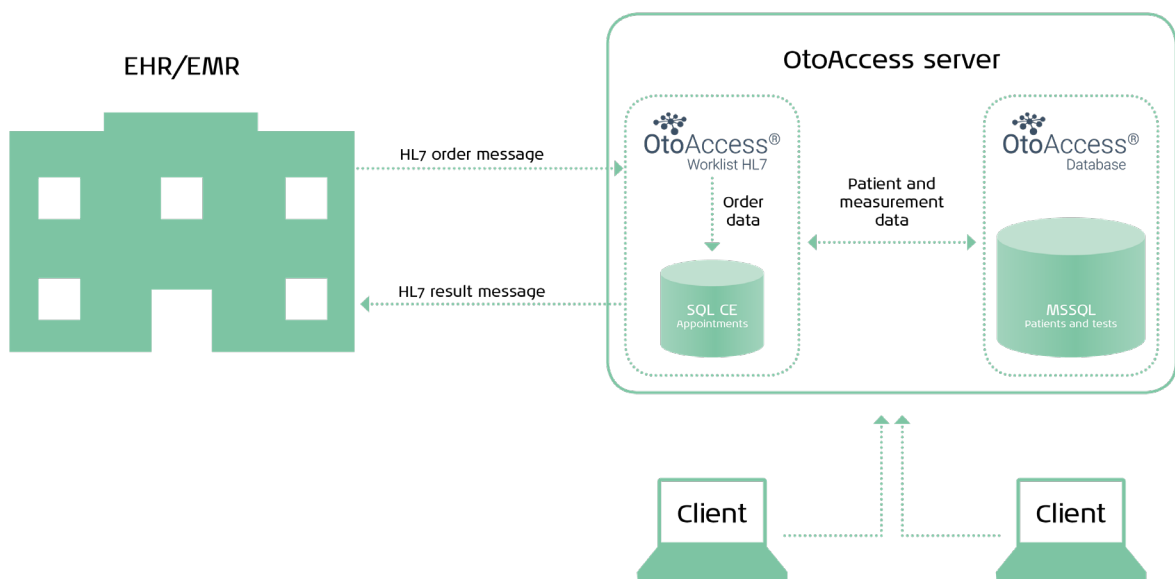


Figure 1

Typically, the OtoAccess® Database and OtoAccess® Worklist HL7 servers are placed on the same Windows® server, but they can be split up if needed.

8.1 Configuration and security

OtoAccess® Worklist HL7 utilizes the MLLP protocol for HL7 communication.

OtoAccess® Worklist HL7 utilizes the communication protocol TLS 1.2 with basic HTTP authentication (shared key) in the communication between the OtoAccess® Worklist HL7 server and clients. See Figure 2 for an illustration.

- The configuration requires a fixed internal IP address and port for the OtoAccess® Worklist HL7 server.
- The configuration requires a fixed IP address and port to the EMR/EHR system.
- The worklist server end point must be set for the clients to find the server.
- The IP-addresses and port numbers can be configured as needed.
- The database password can be configured as needed.
- Category fields can be configured: see section *Searchable Category fields*.
- Policies can be configured: see section *Policies*.



Configuration file example, blue fields are configurable.

```
<?xml version="1.0"?>
<Configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
Version="1">
  <Incoming Endpoint="127.0.0.1:11000" TimeoutInMilliseconds="30001" />
  <Outgoing Endpoint="127.0.0.1:11011" TimeoutInMilliseconds="30001" />
  <Server Endpoint="127.0.0.1:47297" />
  <Database ConnectionString="Data Source=Database.sdf; Password=Ney1reUr2yf8NqUTE90452YnocqsGMpOQLzMqphk; Max
Database Size=4091" />
  <Labels>
    <Category1>Category1</Category1>
    <Category2>Category2</Category2>
  </Labels>
  <Policies>
    <ExpectAckWhenSendingResult IsEnabled="yes"/>
    <PreventResendingResults IsEnabled="no"/>
    <PreventMultipleResults IsEnabled="no"/>
  </Policies>
</Configuration>
```

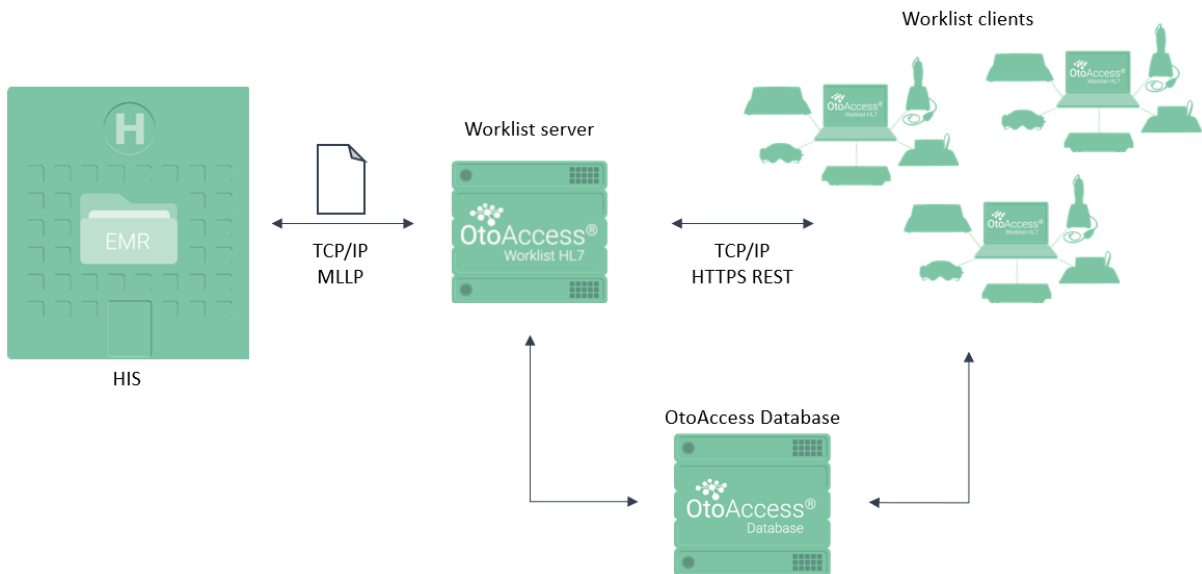


Figure 2

8.2 Searchable Category fields

The Category fields are sorting fields for the user interface. These fields can be named freely in the Configuration.xml file, to accommodate for any needs the clinician may have.

The intent is that these fields will allow you to define how you want to sort your orders. The category fields could be used for e.g. department, filler, modality, building etc.

Note, every time an order is received with a new value in a category field, a new search option will be created in the UI. So be aware to use the Category fields for something purposeful.

8.3 Policies

The Policies section in the Configuration.xml file allows for some key options to be set.

Option	Description	Default value
ExpectAckWhenSendingResult	Defines whether the Worklist server should expect an ACK message after sending a result to the EMR/EHR system or not.	Yes
PreventResendingResults	Defines whether the clinician is prevented to resend an already sent result.	No
PreventMultipleResults	Defines whether the clinician is prevented in creating more than one test result on an order. If not prevented, the clinician can send multiple results on the same order.	No



These policies prevent issues with EMR/EHR systems that cannot handle multiple results on one order or orders being resent.

8.4 Default message formats

The configuration is not locked to specific message types, as it allows for a very flexible setup. The configuration makes the solution able to interpret messages of different types, no matter the message type.

The default order message type is ORM^O01 and the default result message type is ORU^R01. Other functions are by default ADT messages. Only acknowledgement ACK messages are locked to their type.

9 Message configuration

The messages can be configured in the MessageFormat.xml file in the OtoAccess® Worklist HL7 installation folder. See *Appendix 2*.

Any message type can be interpreted. Thus, message names are not required to follow HL7 standards. For example, if you want to send ADT messages instead of ORM messages, this can be set in the MessageFormat.xml configuration file, by changing the conditions.

9.1 Incoming message configuration

Order messages can be configured to perform several functions.

Transaction	Functionality
AddOrder	Will create an order in the OtoAccess® Worklist HL7 user interface. <ul style="list-style-type: none"> If the patient does not exist, the patient will be created in OtoAccess® Database. If the patient already exists, the patient will be updated or used without modification.
UpdateOrder	Will update previously added order. Will also update patient info if changed.
AddOrUpdateOrder	If order exists, it will be updated, otherwise it will be created.
DeleteOrder	Will delete order and order history. Patient and sessions on the patient will not be deleted.
AddPatient	Patient will be created in OtoAccess® Database. No order will be added to Worklist.
UpdatePatient	Patient information will be updated.
DeletePatient	Will delete patient and patient's sessions from OtoAccess® Database as well as deleting any orders from Worklist.

The MessageFormat.xml configuration file in the OtoAccess® Worklist HL7 server installation lets you set conditions for messages and lets you map data fields. See *Appendix 2* for the full configuration.

```
<Filter TransactionType="AddOrder">
  <Conditions>
    <Condition Field="MSH-3-1" Value="HIS" />
    <Condition Field="MSH-4-1" Value="ADM" />
    <Condition Field="MSH-5-1" Value="OTO" />
    <Condition Field="MSH-9-1" Value="ORM" />
    <Condition Field="MSH-9-2" Value="O01" />
    <Condition Field="ORC-1" Value="NW" />
  </Conditions>
  <Maps>
    <Map Field="PID-3-1" Property="Patient.PatientId" />
    <Map Field="PID-5-1" Property="Patient.LastName" />
    <Map Field="PID-5-2" Property="Patient.FirstName" />
  </Maps>
</Filter>
```



```

<Map Field="PID-7-1" Property="Patient.BirthDate:yyyyMMdd" />
<Map Field="OBR-2-1" Property="Order.OrderNumber" />
<Map Field="OBR-4-2" Property="Order.Modality" />
<Map Field="ORC-7-4" Property="Order.ScheduledDate:yyyyMMddHHmm" />
<Map Field="PID-3-1" Property="Order.PatientId" />
<Map Field="PV1-19" Property="Order.VisitNumber" />
<Map Field="OBR-2" Property="Order.Placer" />
<Map Field="OBR-3" Property="Order.Filler" />
    
```

In the above example, only the AddOrder part of MessageFormat.xml is shown.

Each message type consists of **Condition Fields** and **Map Fields**.

- Condition fields lets you define acceptance criteria for the messages. These conditions must be met in full for the message to be accepted and processed.
- Map fields lets you map data fields between the sending and receiving system.

Example – condition fields:

```

<Condition Field="MSH-3-1" Value="HIS" />
    
```

The value in the condition field “MSH-3-1” must be “HIS” to be accepted.

You can choose to change “HIS” to whatever suits your needs, these values are configurable.

Example – map fields:

```

<Map Field="PID-3-1" Property="Patient.PatientId" />
    
```

The contents of “PID-3-1” will be stored as the PatientID in OtoAccess® Worklist HL7.

NOTE: Minimum required fields to map for a message to go through are OrderID and PatientID, the rest are optional.

9.2 Example – AddOrder message

In the above AddOrder example, the conditions are:

HL7 field	Example value	Mandatory	Configurable	Format
MSH-3-1	HIS	Yes	Yes	String
MSH-4-1	ADM		Yes	String
MSH-5-1	OTO	Yes	Yes	String
MSH-9-1	ORM	Yes	Yes	String
MSH-9-2	O01		Yes	String
ORC-1	NW	Yes	Yes	String

And the mappings are

HL7 field	OtoAccess® Worklist HL7 field	Mandatory	Configurable	Format – can be configured
PID-3-1	Patient.PatientId	Yes*	Yes	
PID-5-1	LastName		Yes	
PID-5-2	FirstName		Yes	
PID-7-1	BirthDate:yyyyMMdd		Yes	yyyyMMdd
OBR-2-1	OrderNumber	Yes	Yes	
OBR-4-2	Modality		Yes	
ORC-7-4	ScheduledDate:yyyyMMddHHmm		Yes	yyyyMMddHHmm
PID-3-1	Order.PatientId	Yes*	Yes	
PV1-19	VisitNumber		Yes	
OBR-2	Placer		Yes	
OBR-3	Filler		Yes	

*Typically, same value.



Date and time formats support yyyyMMddHHmmss, e.g. 20210927114537

Syntax	Value	Example
yyyy	Year	2021
MM	Month	09
dd	Day	27
HH	Hour	11
mm	Minute	45
ss	Second	37

Syntax is exact.

You are free to remove or add conditions and mappings and change field mapping to fit your message setup.

NOTE: For the full list of supported condition and patient fields, see Appendix 1.

9.3 Other incoming message types

We have several types of incoming messages. They are either Order, Order/Patient or Patient related. Refer to the section 'Basic solution infrastructure'.

9.4 Order related messages

By default, orders are ORM^O01 messages. The ORC-1 field is used to decide the type:

Message function	ORC-1 default value*
AddOrder	NW
UpdateOrder	XO
AddOrUpdateOrder	NWXO
DeleteOrder	OC

*Default values can be configured as needed or defines by other fields.

9.4.1 Patient only related messages

Because patient information is stored in OtoAccess® Database, you have the option to add, update or delete patient directly in the database via HL7 messages.

This is controlled by message types:

Function	Default message type
AddPatient	ADT^A04
UpdatePatient	ADT^A08
DeletePatient	ADT^A23

Conditions and mapping can be adjusted as needed.



Example: AddPatient message

```
MSH|^~\&|HIS|ADM|OTO||20190715130450||ADT^A04|fW9Bf4KahqjjcYsJjnPcdJTj|T|2.4
PID|||12457885-1254|Brown^Peter||20120715
```

Extract from the MessageFormat.xml file, that shows conditions and mappings for the AddPatient message. The colors highlight the connection between the .xml file and the AddPatient message.

See Appendix 1 for all available fields.

```
</Filter>
  <Filter TransactionType="AddPatient">
    <Conditions>
      <Condition Field="MSH-3-1" Value="HIS" />
      <Condition Field="MSH-4-1" Value="ADM" />
      <Condition Field="MSH-5-1" Value="OTO" />
      <Condition Field="MSH-9-1" Value="ADT" />
      <Condition Field="MSH-9-2" Value="A04" />
    </Conditions>
    <Maps>
      <Map Field="PID-3-1" Property="Patient.PatientId" />
      <Map Field="PID-5-1" Property="Patient.LastName" />
      <Map Field="PID-5-2" Property="Patient.FirstName" />
      <Map Field="PID-7-1"
Property="Patient.BirthDate:yyyyMMdd" />
    </Maps>
```

Example: DeletePatient message

```
MSH|^~\&|HIS|ADM|OTO||20190715131831||ADT^A23|rL9F68Qm5B4djR2BsNF6jZMx|T|2.7.1
PID|||12457885-1254
```

Extract from the MessageFormats.xml file, that shows conditions and mappings for the DeletePatient message. The color highlight the connection between the .xml file and the DeletePatient message.

```
<Filter TransactionType="DeletePatient">
  <Conditions>
    <Condition Field="MSH-3-1" Value="HIS" />
    <Condition Field="MSH-4-1" Value="ADM" />
    <Condition Field="MSH-5-1" Value="OTO" />
    <Condition Field="MSH-9-1" Value="ADT" />
    <Condition Field="MSH-9-2" Value="A23" />
  </Conditions>
  <Maps>
    <Map Field="PID-3-1" Property="Patient.PatientId" />
  </Maps>
```



10 Outgoing message configuration

In the MessageFormat.xml configuration file you can set up the outgoing message to fit your needs.

Below, the extract from the configurations show the default setup and below that the resulting message.

Configuration example

```
<Outgoing IsAckExpected="true">
  <Templates>
    <Template>
      <Maps>
        <Map Field="MSH-1" Property="|" />
        <Map Field="MSH-2" Property="^~\&" />
        <Map Field="MSH-3-1" Property="WorkLIST" />
        <Map Field="MSH-5-1" Property="HIS" />
        <Map Field="MSH-7-1" Property="\$(Environment.Timestamp:yyyyMMddhhmmss)"
/ >

        <Map Field="MSH-9-1" Property="ORU" />
        <Map Field="MSH-9-2" Property="R01" />
        <Map Field="MSH-10" Property="MSG-\$(Environment.Timestamp:yyMMdd)-
\$(Environment.Timestamp:ss)" />
        <Map Field="MSH-11-1" Property="D" />
        <Map Field="MSH-12-1" Property="2.3.1" />
        <Map Field="PID-3-1" Property="\$(Patient.PatientId)" />
        <Map Field="PID-5-1-1" Property="\$(Patient.LastName)" />
        <Map Field="PID-5-2" Property="\$(Patient.FirstName)" />
        <Map Field="PID-7-1" Property="\$(Patient.BirthDate)" />
        <Map Field="PV1-19-1" Property="\$(Order.VisitNumber)" />
        <Map Field="OBR-3-1" Property="\$(Order.Filler)" />
        <Map Field="OBR-34-1-1" Property="\$(Result.Examiner)" />
        <Map Field="OBX-1" Property="1" />
        <Map Field="OBX-2" Property="ED" />
        <Map Field="OBX-4" Property="RTXX" />
        <Map Field="OBX-5-5" Property="\$(Result.Report)" />
        <Map Field="OBX-11" Property="F" />
      </Maps>
    </Template>
  </Templates>
</Outgoing>
```

Output example

```
MSH|^~\&|WorkLIST||HIS||20190715013146||ORU^R01|MSG-190715-46|D|2.3.1
PID|||10373629||Hansson^Pete||03-11-1965 00:00:00
PV1|||||||||||||||||1
OBR|||Dr. Demant|||||||||||||||||||||||||
OBX|1|ED|RTXX|^|^^(RESULT BASE64 encoded*)|||F
```

*Result will be appended Base64-encoded or the message will contain a file path to the result.



10.1 Result setup and format

*You can choose **pdf** format of Report/Image by specifying it after colon (default is **xps**). In this case, the result will be appended to the outgoing HL7 message Base-64 encoded.

```
<OutgoingMap DestinationField="OBX-5-5" SourceProperty="$(Result.Report:pdf)" />
<OutgoingMap DestinationField="OBX-5-5" SourceProperty="$(Result.Image)" />
```

10.2 Result delivered to file share

As an addition or alternative to having the result appended to the result message, the result can be placed in a file share.

```
<FileDrops>
  <FileDrop Path="c:\temp\Report $(Patient.PatientId).pdf" Content="$(Result.Report:pdf)" />
</FileDrops>
```

In the above example, the file will be 'Result PatientID.pdf' e.g. 'Report 6392936.pdf'.

You have full flexibility in setting up the naming using standard XML syntax.

11 Message examples

11.1 Order messages

AddOrder example

```
MSH|^~\&|HIS|ADM|OTO||20190722140144||ORM^001|jnE2NwWDAMpBs5EKtNUJP7Ra|T|2.4
PID|||078-05-1120||Smith^John||19820517
PV1|||||||||||||||||||||1
ORC|NW|||^^^202008221030
OBR||2|Dr. Demant|^VNG
```

UpdateOrder example

```
MSH|^~\&|HIS|ADM|OTO||20190722140144||ORM^001|jnE2NwWDAMpBs5EKtNUJP7Ra|T|2.4
PID|||078-05-1120||Smith^John||19820517
PV1|||||||||||||||||||||1
ORC|XO|||^^^202009131145
OBR||2|Dr. Demant|^VNG
```

AddOrUpdateOrder example

```
MSH|^~\&|HIS|ADM|OTO||20190722140144||ORM^001|jnE2NwWDAMpBs5EKtNUJP7Ra|T|2.4
PID|||078-05-1120||Smith^John||19820517
PV1|||||||||||||||||||||1
ORC|NWXO|||^^^202009141100
OBR||2|Dr. Demant|^VNG
```



DeleteOrder example

```
MSH|^~\&|HIS|ADM|OTO||20190722140144||ORM^O01|jnE2NwWDAMpBs5EKtNUJP7Ra|T|2.4
ORC|OC
OBR||2
```

11.2 Patient messages

AddPatient example

```
MSH|^~\&|HIS|ADM|OTO||20190722141957||ADT^A04|aFNGPWAIQmbuTr97NIhUaWGs|T|2.7.1
PID|||078-05-1121||Anderson^George||19960718
```

UpdatePatient example

```
MSH|^~\&|HIS|ADM|OTO||20190722141957||ADT^A08|aFNGPWAIQmbuTr97NIhUaWGs|T|2.7.1
PID|||078-05-1121||Peterson^George||19960718
```

DeletePatient

```
MSH|^~\&|HIS|ADM|OTO||20190722141957||ADT^A23|aFNGPWAIQmbuTr97NIhUaWGs|T|2.7.1
PID|||078-05-1121
```

11.3 Result messages

Result message in default configuration

```
MSH|^~\&|WorkLIST||HIS||20190722024945||ORU^R01|MSG-190722-45|D|2.3.1
PID|||078-05-1120||Smith^John||17-05-1982 00:00:00
PV1|||||||||||||||||||||1
OBR|||Dr. Demant|||||||||||||||||||||||||||||||||
OBX|1|ED||RTXX|^XPS/PDF Report/Image base64 encoded|||||F
```

Included result optional, can also be placed in shared folder.

11.4 Using ADT instead of ORM example

Example – using ADT^01 instead of ORM^O01

```
MSH|^~\&|HIS|ADM|OTO||20190722135659||ADT^A01|70wmHfnE3Yn4URcXj627xzmS|T|2.4
PID|||365478-1125||Nilsson^Peter||19540812
PV1|||||||||||||||||||||1
ORC|NW|||||^201903221030
OBR||10345|Dr. Demant|^AUD
```



Checklist

Please mark and sign if read and accepted.

- Chapter 6: Prerequisites are acceptable.
- Chapter 7: Workflow is acceptable.
- Chapter 8: Configuration and security is acceptable.
- Chapter 9 to 11: Message options are acceptable

Date:

For customer

Name in caps:

Company/institution:



Appendix 1: Available message fields

Supported message fields, the patient fields are reflected in OtoAccess® Database.

Order

- OrderNumber
- PatientId
- Category1
- Category2
- ScheduledDate*

Patient

- PatientId
- BirthDate
- ExpectedBirthDate
- FirstName
- LastName
- Address
- Address2
- City
- ZipCode
- County
- Physician
- Mobile
- Telephone
- Email
- Insurer
- InsuranceNumber
- IsInsured**
- ReferredBy
- Initials
- Municipal
- State
- CountryLCID
- Fax
- MothersName
- Occupation
- Remarks
- BillingNumber

*For dates, you can use specific formats by using colon:

e.g.

```
<Map Field="ORC-7-4" Property="Order.ScheduledDate:yyyyMMddHHmm" />
```

**It is Boolean, only true/false values are allowed.



Appendix 2: Messageformats.xml configuration file

Below is the default configuration file, the file can be changed to match your configuration needs. Changes will take effect after restart of the OtoAccess® Worklist HL7 server service.

```
<?xml version="1.0"?>
<Configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" Version="1">
  <!--HL7 Message encoding. Default codepage: 65001 = Unicode (UTF-8)-->
  <Encoding Codepage="65001" />
  <Incoming>
    <!--Supported messages

    AddOrder      -Will create an order in the Worklist HL7 user interface.
                  -If the patient does not exist, a new patient record will be created in the OtoAccess Database.
                  -If the patient already exists it will be updated if needed or used without modification.

    UpdateOrder   -Will update a previously added order, and also update the patient if it is changed.
    AddOrUpdateOrder -If the order exists it'll be updated, otherwise it will be created.
    DeleteOrder   -Will delete an order and its history. Patient and sessions will not be affected.
    AddPatient    -A Patient record will be created in OtoAccess Database. Nothing will be added to Worklist.
    UpdatePatient -Patient information will be updated.
    DeletePatient -Will delete everything related to the patient in OtoAccess Database and OtoAccess Worklist HL7.

    Result Message -Can be configured as needed.
  -->

  <!--AddOrder-->
  <Filter TransactionType="AddOrder">
    <Conditions>
      <Condition Field="MSH-3-1" Value="HIS" />
      <Condition Field="MSH-4-1" Value="ADM" />
      <Condition Field="MSH-5-1" Value="OTO" />
      <Condition Field="MSH-9-1" Value="ORM" />
      <Condition Field="MSH-9-2" Value="O01" />
      <Condition Field="ORC-1" Value="NW" />
    </Conditions>
    <Maps>
      <IncomingMap SourceField="PID-3-1" DestinationProperty="Patient.PatientId" />
      <IncomingMap SourceField="PID-5-1" DestinationProperty="Patient.LastName" />
      <IncomingMap SourceField="PID-5-2" DestinationProperty="Patient.FirstName" />
      <IncomingMap SourceField="PID-7-1" DestinationProperty="Patient.BirthDate:yyyyMMdd" />
      <IncomingMap SourceField="OBR-2-1" DestinationProperty="Order.OrderNumber" />
      <IncomingMap SourceField="OBR-4-2" DestinationProperty="Order.Category1" />
      <IncomingMap SourceField="ORC-7-4" DestinationProperty="Order.ScheduledDate:yyyyMMddHHmm" />
      <IncomingMap SourceField="PID-3-1" DestinationProperty="Order.PatientId" />
      <IncomingMap SourceField="PV1-19" DestinationProperty="Order.Text2" />
      <IncomingMap SourceField="OBR-2" DestinationProperty="Order.Text1" />
      <IncomingMap SourceField="OBR-3" DestinationProperty="Order.Category2" />
    </Maps>
  </Filter>

  <!--UpdateOrder-->
  <Filter TransactionType="UpdateOrder">
    <Conditions>
      <Condition Field="MSH-3-1" Value="HIS" />
      <Condition Field="MSH-4-1" Value="ADM" />
      <Condition Field="MSH-5-1" Value="OTO" />
      <Condition Field="MSH-9-1" Value="ORM" />
      <Condition Field="MSH-9-2" Value="O01" />
      <Condition Field="ORC-1" Value="XO" />
    </Conditions>
    <Maps>
      <IncomingMap SourceField="OBR-2-1" DestinationProperty="Order.OrderNumber" />
      <IncomingMap SourceField="OBR-4-2" DestinationProperty="Order.Category1" />
      <IncomingMap SourceField="ORC-7-4" DestinationProperty="Order.ScheduledDate:yyyyMMddHHmm" />
      <IncomingMap SourceField="PID-3-1" DestinationProperty="Order.PatientId" />
      <IncomingMap SourceField="PV1-19" DestinationProperty="Order.Text2" />
      <IncomingMap SourceField="OBR-2" DestinationProperty="Order.Text1" />
      <IncomingMap SourceField="OBR-3" DestinationProperty="Order.Category2" />
      <IncomingMap SourceField="PID-3-1" DestinationProperty="Patient.PatientId" />
      <IncomingMap SourceField="PID-5-1" DestinationProperty="Patient.LastName" />
      <IncomingMap SourceField="PID-5-2" DestinationProperty="Patient.FirstName" />
      <IncomingMap SourceField="PID-7-1" DestinationProperty="Patient.BirthDate:yyyyMMdd" />
    </Maps>
  </Filter>

  <!--AddOrUpdateOrder-->
  <Filter TransactionType="AddOrUpdateOrder">
    <Conditions>
      <Condition Field="MSH-3-1" Value="HIS" />
      <Condition Field="MSH-4-1" Value="ADM" />
      <Condition Field="MSH-5-1" Value="OTO" />
      <Condition Field="MSH-9-1" Value="ORM" />
      <Condition Field="MSH-9-2" Value="O01" />
      <Condition Field="ORC-1" Value="NWXO" />
    </Conditions>
    <Maps>
      <IncomingMap SourceField="OBR-2-1" DestinationProperty="Order.OrderNumber" />
      <IncomingMap SourceField="OBR-4-2" DestinationProperty="Order.Category1" />
      <IncomingMap SourceField="ORC-7-4" DestinationProperty="Order.ScheduledDate:yyyyMMddHHmm" />
      <IncomingMap SourceField="PID-3-1" DestinationProperty="Order.PatientId" />
      <IncomingMap SourceField="PV1-19" DestinationProperty="Order.Text2" />
      <IncomingMap SourceField="OBR-2" DestinationProperty="Order.Text1" />
    </Maps>
  </Filter>

```



```

<IncomingMap SourceField="OBR-3" DestinationProperty="Order.Category2" />
<IncomingMap SourceField="PID-3-1" DestinationProperty="Patient.PatientId" />
<IncomingMap SourceField="PID-5-1" DestinationProperty="Patient.LastName" />
<IncomingMap SourceField="PID-5-2" DestinationProperty="Patient.FirstName" />
<IncomingMap SourceField="PID-7-1" DestinationProperty="Patient.BirthDate:yyyyMMdd" />
</Maps>
</Filter>

<!--DeleteOrder-->
<Filter TransactionType="DeleteOrder">
  <Conditions>
    <Condition Field="MSH-3-1" Value="HIS" />
    <Condition Field="MSH-4-1" Value="ADM" />
    <Condition Field="MSH-5-1" Value="OTO" />
    <Condition Field="MSH-9-1" Value="ORM" />
    <Condition Field="MSH-9-2" Value="O01" />
    <Condition Field="ORC-1" Value="OC" />
  </Conditions>
  <Maps>
    <IncomingMap SourceField="OBR-2-1" DestinationProperty="Order.OrderNumber" />
  </Maps>
</Filter>

<!--AddPatient-->
<Filter TransactionType="AddPatient">
  <Conditions>
    <Condition Field="MSH-3-1" Value="HIS" />
    <Condition Field="MSH-4-1" Value="ADM" />
    <Condition Field="MSH-5-1" Value="OTO" />
    <Condition Field="MSH-9-1" Value="ADT" />
    <Condition Field="MSH-9-2" Value="A04" />
  </Conditions>
  <Maps>
    <IncomingMap SourceField="PID-3-1" DestinationProperty="Patient.PatientId" />
    <IncomingMap SourceField="PID-5-1" DestinationProperty="Patient.LastName" />
    <IncomingMap SourceField="PID-5-2" DestinationProperty="Patient.FirstName" />
    <IncomingMap SourceField="PID-7-1" DestinationProperty="Patient.BirthDate:yyyyMMdd" />
  </Maps>
</Filter>

<!--UpdatePatient-->
<Filter TransactionType="UpdatePatient">
  <Conditions>
    <Condition Field="MSH-3-1" Value="HIS" />
    <Condition Field="MSH-4-1" Value="ADM" />
    <Condition Field="MSH-5-1" Value="OTO" />
    <Condition Field="MSH-9-1" Value="ADT" />
    <Condition Field="MSH-9-2" Value="A08" />
  </Conditions>
  <Maps>
    <IncomingMap SourceField="PID-3-1" DestinationProperty="Patient.PatientId" />
    <IncomingMap SourceField="PID-5-1" DestinationProperty="Patient.LastName" />
    <IncomingMap SourceField="PID-5-2" DestinationProperty="Patient.FirstName" />
    <IncomingMap SourceField="PID-7-1" DestinationProperty="Patient.BirthDate:yyyyMMdd" />
  </Maps>
</Filter>

<!--DeletePatient-->
<Filter TransactionType="DeletePatient">
  <Conditions>
    <Condition Field="MSH-3-1" Value="HIS" />
    <Condition Field="MSH-4-1" Value="ADM" />
    <Condition Field="MSH-5-1" Value="OTO" />
    <Condition Field="MSH-9-1" Value="ADT" />
    <Condition Field="MSH-9-2" Value="A23" />
  </Conditions>
  <Maps>
    <IncomingMap SourceField="PID-3-1" DestinationProperty="Patient.PatientId" />
  </Maps>
</Filter>
</Incoming>
<Outgoing>
<!--Outgoing message configuration-->

<!--Result message-->
<Template Name="Result">
  <Maps>
    <OutgoingMap DestinationField="MSH-1" SourceProperty="|" />
    <OutgoingMap DestinationField="MSH-2" SourceProperty="^~&" />
    <OutgoingMap DestinationField="MSH-3-1" SourceProperty="Worklist" />
    <OutgoingMap DestinationField="MSH-5-1" SourceProperty="HIS" />
    <OutgoingMap DestinationField="MSH-7-1" SourceProperty="{(Environment.Timestamp:yyyyMMddhhmmss)}" />
    <OutgoingMap DestinationField="MSH-9-1" SourceProperty="ORU" />
    <OutgoingMap DestinationField="MSH-9-2" SourceProperty="R01" />
    <OutgoingMap DestinationField="MSH-10" SourceProperty="MSG-{(Environment.Timestamp:yyMMdd)}-{(Environment.Timestamp:ss)}" />
    <OutgoingMap DestinationField="MSH-11-1" SourceProperty="D" />
    <OutgoingMap DestinationField="MSH-12-1" SourceProperty="2.3.1" />
    <OutgoingMap DestinationField="PID-3-1" SourceProperty="{(Patient.PatientId)}" />
    <OutgoingMap DestinationField="PID-5-1-1" SourceProperty="{(Patient.LastName)}" />
    <OutgoingMap DestinationField="PID-5-2" SourceProperty="{(Patient.FirstName)}" />
    <OutgoingMap DestinationField="PID-7-1" SourceProperty="{(Patient.BirthDate)}" />
    <OutgoingMap DestinationField="PV1-19-1" SourceProperty="{(Order.Text2)}" />
    <OutgoingMap DestinationField="OBR-3-1" SourceProperty="{(Order.Category2)}" />
    <OutgoingMap DestinationField="OBR-34-1-1" SourceProperty="{(Result.Examiner)}" />
    <OutgoingMap DestinationField="OBX-1" SourceProperty="1" />
    <OutgoingMap DestinationField="OBX-2" SourceProperty="ED" />
    <OutgoingMap DestinationField="OBX-4" SourceProperty="RTXX" />
  </Maps>

```




```
<OutgoingMap DestinationField="OBX-5-5" SourceProperty="{Result.Report}" />
<OutgoingMap DestinationField="OBX-11" SourceProperty="F" />
</Maps>
<FileDrops>
  <FileDrop Path="c:\temp\Report ${Patient.PatientId}.pdf" Content="{Result.Report:pdf}" />
</FileDrops>
</Template>
</Outgoing>
</Configuration>
```

