



Outbound Patients, Appointments and Inbound Charges

Common Use Case Package

athenahealth, Inc.

Updated: October 2020

Formerly EMR Lite and EMR Full

Table of Contents

[1 Introduction 3](#_Toc45615269)

[1.1 Interface Description 3](#_Toc45615270)

[1.2 Scope Overview 3](#_Toc45615271)

[1.3 Scope Process 3](#_Toc45615272)

[2 Project Information 4](#_Toc45615273)

[3 Common and Specific Use Cases 5](#_Toc45615274)

[4 Interface Enablement 6](#_Toc45615275)

[5 Interface Configuration 7](#_Toc45615276)

[5.1 Message Samples and Specifications 7](#_Toc45615277)

[5.2 Integration Testing 7](#_Toc45615278)

[5.2.1 Testing Phases 7](#_Toc45615279)

[5.3 Connectivity Details 7](#_Toc45615280)

[5.4 Backfills 7](#_Toc45615281)

[5.4.1 Backfills via the Interface 7](#_Toc45615282)

[5.5 External Identity Management 8](#_Toc45615283)

[6 Outbound Message Configuration 9](#_Toc45615284)

[6.1 Message Filtering 9](#_Toc45615285)

[6.1.1 Selective Filtering of Outbound Messages 9](#_Toc45615286)

[6.2 Patients 9](#_Toc45615287)

[6.2.1 Patient Race, Ethnicity, and Language 9](#_Toc45615288)

[6.3 Appointments 9](#_Toc45615289)

[6.3.1 Appointment Statuses 10](#_Toc45615290)

[6.3.2 Provider ID Management 10](#_Toc45615291)

[7 Inbound Message Configuration 11](#_Toc45615292)

[7.1 Charges 11](#_Toc45615293)

[7.1.1 Minimum Required Fields for Charge Messages 11](#_Toc45615294)

[7.1.2 Matching Logic for Charge Messages 11](#_Toc45615295)

[7.1.3 Processing Logic for Charge Messages 12](#_Toc45615296)

[8 Scope Approval 14](#_Toc45615297)

[8.1 Additional Comments 14](#_Toc45615298)

[9 Appendices and Other References 15](#_Toc45615299)

[9.1 Interface Message Queue Manager 15](#_Toc45615300)

[9.2 Continuing Service and Support 15](#_Toc45615301)

1. Introduction

This document provides information for an interface that supports the following data exchange:

* Outbound patient messages
* Outbound appointment messages
* Inbound Charges

Your organization may not have requested each integration; athenahealth specifies the sections you can skip if they’re not applicable.

* 1. Interface Description

This interface supports the secure and automated transfer of information between athenaNet and an external third-party system. Interface data is formatted according to HL7 v2 standards to ensure compatibility across a wide array of platforms and software vendors. Interface data may include:

* External patient identifiers (e.g., a medical record number (MRN) assigned by a third-party vendor system)
* Patient demographics (e.g., name, date of birth, address, and so on)
* Patient insurance (e.g., carrier, member ID, and so on)
* Charge data (e.g., diagnosis codes, CPT codes, date of service, and so on)
  1. Scope Overview

This is a pre-scoped standard interface, which means athenahealth has selected many of the configurations for your convenience. If you require customization to this integration outside of what this document provides, contact your athenahealth Interface Project Engineer and they’ll connect you with the athenahealth Integration Design team for more detailed scoping. Please note that customizing the integration may incur fees.

* 1. Scope Process

1. **Review the project** – Read the entire Common Use Case Package
2. **Enter or select required information to configure the interface** –
   1. Double-click the gray fields and boxes that appear in the tables and within the text.  
      The Form Field Options window opens.
   2. For fields, enter the information in the Default Text Field. For checkboxes, select Checked or Unchecked. For menus, select the option in the Items in Drop-Down list box.
   3. Click OK.
3. **Approve the project** – Enter your name and date in the Scope Approval section to approve the scope of the interface on page 14.
4. **Return the completed CUC scope document as a Word doc**– this doesn’t require a wet signature and shouldn’t be returned as a PDF.

REMEMBER: Your athenahealth Interface Project Engineer is available to meet, assist with questions, and help you scope the project to determine the best options for your organization.

1. Project Information

Table 1 - General information

|  |  |
| --- | --- |
|  |  |
| athenahealth practice context ID |  |
| athenahealth practice name |  |
| Event number (provided by Interface Project Engineer for internal athenahealth tracking) |  |
| Vendor name |  |
| Vendor type (e.g., health information system, electronic health record, and so on.) |  |

1. Common and Specific Use Cases

It’s important to understand the related workflows and how this interface will exchange data between athenaNet and the third-party vendor system in support of those workflows.

Review the common use cases described in the table and think about how your organization will use the interface.

Table 2 - Interface use cases

|  |  |  |
| --- | --- | --- |
| Use Case | Event | Functionality |
| Patient synchronization | New Patient ADDED in athenaNet  Patient UPDATED in athenaNet | Patient ADDED in other system  Patient UPDATED in other system |
| Schedule synchronization | Appt SCHEDULED in athenaNet  Appt CANCELLED in athenaNet  Appt CHECKIN in athenaNet  Appt CHECKOUT in athenaNet  Appt UPDATE in athenaNet | Appt SCHEDULED in other system  Appt CANCELLED in other system  Appt status UPDATED in other system  Appt status UPDATED in other system  Appt UPDATED in other system |
| Single charge message | Charge finalized in other system | A claim is created in athenaNet. |

 Think about your how organization will use the interface. Use these questions to help guide you:

* How does the third-party vendor system fit into our user workflows in athenaNet?
* Does the third-party vendor need athenaNet appointment data?

**Organization’s specific use cases and workflow description:**

**TIP**: Review common and specific use cases with your athenahealth Interface Project Engineer until you’re comfortable with the intended functionality. This ensures that you can prepare staff for changes to their workflow (e.g., parts of their workflow that are automated versus manual) that often occur with the introduction of a new interface.

1. Interface Enablement

Select the configurations your organization wants to enable for the interface. Check the box in the **Enable** column of table 3 to make a selection.

Table 3 - Interface enablement

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Enable | Action | Event | Direction | Default interface message |
|  | Add Patient | New Patient ADDED in athenaNet | Outbound | A28 |
|  | Update Patient | New Patient UPDATED in athenaNet | Outbound | A31 |
|  | Schedule Appointment | Appt SCHEDULED in athenaNet | Outbound | S12 |
|  | Update Appointment | Appt UPDATE in athenaNet | Outbound | S14 |
|  | Cancel Appointment | Appt CANCELLED in athenaNet | Outbound | S15 |
|  | Check-In | Appt CHECKIN in athenaNet | Outbound | S14 |
|  | Check-Out | Appt CHECKOUT in athenaNet | Outbound | S14 |
|  | Charges | Charges FINALIZED in other system | Inbound | P03 |

1. Interface Configuration
   1. Message Samples and Specifications

See the HL7v2 section of the [Developer Toolkit](http://www.athenahealth.com/developer-portal/developer-toolkit/by-standard) (http://www.athenahealth.com/developer-portal/developer-toolkit/by-standard) to review message samples and specifications.

* 1. Integration Testing

athenahealth provides a non-live, athenahealth-hosted test environment (“Preview”) to facilitate integration testing before moving the interface to production. You should expect the third-party vendor to provide a similar non-live testing environment.

Will the third-party vendor provide a testing environment for this project? Yes is recommended

If you answered “No,” explain how you will test the integration on the third-party vendor system:

* + 1. Testing Phases

Interface testing generally occurs in two phases: unit testing and end-user testing.

1. **Unit testing phase**athenahealth works directly with the third-party vendor to ensure that messages are being triggered, sent, received, and processed successfully in the respective system. During this phase your organization may be asked to confirm that the data in either system are accurate.
2. **End-user testing phase**  
   The end-user testing phase begins after the unit testing phase. athenahealth provides general test plans and your organization plans, organizes, and executes interface testing as it relates to practice workflows. athenahealth may provide guidance when appropriate.

**BEST PRACTICE:** athenahealth recommends creating test plans specific to practice workflows, in addition to those athenahealth provides, for a more comprehensive end-user testing phase.

* 1. Connectivity Details

As part of interface implementation, athenahealth needs to establish a secure method of transfer for electronic data to and from a third-party system. The Connectivity Method Overview document contains our current connectivity options and information regarding functionality and project steps.

<http://www.athenahealth.com/~/media/athenaweb/files/developer-portal/Connectivity_Methods_Overview.docx>

Contact your athenahealth Interface Project Engineer if you have questions.

* 1. Backfills
     1. Backfills via the Interface

Your organization can request a backfill, where athenaNet sends a full load of all patient and future appointment data between your organization and the third-party vendor as the interface is first enabled.

If you require a backfill, it’s important to consider the complexities of integrating data from several different systems. For example, when athenaNet backfills information received from a third-party vendor system, the data often includes foreign IDs that need to be added to your custom fields. The IDs must be unique and might be bi-directionally accepted between all systems.

Does this project require a backfill?

Additional comments:

* 1. External Identity Management

To assist with ID management throughout an integrated health system, athenaNet can store multiple external IDs, at both the patient and appointment level. External IDs may be used for matching purposes or external IDs may just be interfaced and stored in athenaNet using custom fields. All external IDs present in athenaNet, such as those supplied by an interface or import process, are available to be sent out over the interface.

If external IDs are included in scope for your integration, please complete tables 4 and 5 with your athenahealth resource. Enter the name and ID of each patient-level and appointment-level custom field.

**REMEMBER:** You can match only one external ID per record number category even if you receive multiple IDs. athenaNet assumes the external ID is correct, therefore external IDs must be unique and non-changing.

|  |  |  |
| --- | --- | --- |
| athenaNet custom field name | athenaNet custom field ID | HL7 field |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

*Table 4 - Patient-level custom fields*

Table 5 – Appointment-level external IDs fields

|  |  |  |
| --- | --- | --- |
| athenaNet External Appointment ID field name | athenaNet custom field ID | HL7 field |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Are any of the above external IDs formatted with leading zeros?

By default, the information in the above tables is applied to all outbound messages when available.

Additional comments:

1. Outbound Message Configuration

This section contains outbound message configurations for:

* Patients
* Appointments
  1. Message Filtering
     1. Selective Filtering of Outbound Messages

You can filter outbound messages, so the interface sends them only for particular providers or departments.

Should the interface filter outbound messages? No is recommended (i.e., the interface sends all configured messages)

If you answered “Yes,” complete table 6. Select the entity in the **Filter By** column and enter the name or names of the entity you want to filter by in the **Names** column.

Table 6 - Filter outbound messages

|  |  |  |  |
| --- | --- | --- | --- |
| Message Type | Filter By |  | Names |
| **Patients** |  | Provider |  |
|  |  | Department |  |
|  |  | Provider Group |  |
|  |  | Appointment Type |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Message Type | Filter By |  | Names |
| **Appointments** |  | Provider |  |
|  |  | Department |  |
|  |  | Provider Group |  |
|  |  | Appointment Type |  |

* 1. Patients

This subsection provides configurations for outbound patient messages. **Please skip to subsection 6.3 if outbound patient messages are out of scope.**

* + 1. Patient Race, Ethnicity, and Language

For outbound patient messages, the interface sends race, ethnicity and language in the below formats:

* Race/Ethnicity: CDC identifier (e.g., “1019-9” for the While Mountain Apache race)
* Language: ISO6392 code (e.g., “eng” for English)
  1. Appointments

This subsection provides configurations for outbound appointment messages. **Please skip to section 7 if outbound appointment messages are out of scope.**

* + 1. Appointment Statuses

For outbound appointment messages, the interface sends the appointment status in field SCH.25.  The statuses coincide with the event that triggered the message by default. Review the statuses in table 7.

Table 7 - Default appointment statuses

|  |  |
| --- | --- |
| Event | SCH.25 value |
| New appointment | BOOKED |
| Appointment check-in | ARRIVED |
| Appointment checkout | COMPLETED |
| Appointment update | UPDATE |
| Cancel appointment | CANCELLED |

DID YOU KNOW?: When a user reschedules an appointment through the athenaNet appointment workflow, athenaNet cancels the original appointment record and creates a new appointment record with a new athenaNet appointment ID. The interface generates an appointment cancel message for the original appointment and an appointment create message for the new appointment. Contact your athenahealth Interface Project Engineer if this functionality will be an issue for your downstream system.

* + 1. Provider ID Management

The interface configures the provider ID in outbound messages as either the provider’s National Provider Number (NPI) or the athenaNet provider ID.Select your preferred provider ID configuration in table 8.

Table 8 - Provider ID configuration options

|  |  |
| --- | --- |
|  | Option |
|  | NPI |
|  | athenaNet provider ID |

1. Inbound Message Configuration

This section provides inbound message configurations for:

* Charges
  1. Charges

This subsection provides configurations for inbound charge messages. **Please skip to section 8 if inbound charge messages are out of scope.**

* + 1. Minimum Required Fields for Charge Messages

To create a claim, athenaNet requires the third-party vendor system to populate messages with the HL7 fields specified in table 9. Review the required HL7 fields with the third-party vendor.

Table 9 - Required HL7 Fields To Create A Claim

|  |  |
| --- | --- |
| Data field | Default HL7 field |
| Rendering Provider | Derived from Appointment or FT1.20 |
| Supervising Provider | Derived from the Rendering Provider or FT1.21 |
| Department | Derived from Appointment or FT1.16 or FT1.13 |
| Service Date | Derived from Appointment or FT1.4 |
| Procedure Code | FT1.25 |
| Modifier (if required for procedure code) | FT1.26 |
| Diagnosis Code | FT1.19 |

REMEMBER: Only charge data is processed from the inbound charge message. All other data in the DFT, including demographic updates, are discarded.

BEST PRACTICE: The third-party vendor system should only send DFT messages when the charge data are complete, finalized, and ready for immediate billing out of athenaNet. They should send all charges at once, ideally contained within single DFT messages (one claim per message). athenahealth does not support “building up a claim” over the course of many transactions, charges, or interface messages.

* + 1. Matching Logic for Charge Messages
       1. Patient Matching for Charge Messages

For this interface, the athenaNet patient matching algorithm compares demographic information in athenaNet with the data elements in each message received. The data elements used for patient matching are athena patient ID, client-specified external patient ID, full last name, full first name, date of birth, SSN, gender, middle initial, address and phone number. The athenaNet Interface Message Queue Manager provides a manual review process for messages that may create duplicate patient records or substantially change the demographics for an existing patient record.

* + - 1. Appointment Matching for Charge Messages

athenaNet can match charges to an appointment using either the athenaNet appointment ID or an external appointment ID. Using the athenaNet ID assumes that the third-party vendor system is receiving and storing these IDs and has them available to send, whereas using an external appointment ID assumes that the third-party vendor system has previously sent the ID and athenaNet has stored it on the appointment. When a DFT message matches to an appointment, athenaNet moves the appointment status from 2 (checked in) or 3 (checked out) to 4 (charges entered). Select your organization’s preferred method of matching inbound charge messages to appointments in Table 10.Additionally, select the HL7 field where the appointment ID information will appear.

Table 10 - Matching Charge Messages to Appointments Options

|  |  |
| --- | --- |
|  | Option |
|  | **athenaNet matches charges to an appointment using the athenaNet appointment ID in the inbound charge message, when it’s available.**  In which field will the appointment ID appear? |
|  | **athenaNet matches charges to an appointment using the external appointment ID in the inbound charge message, when it’s available.**  In which field will the appointment ID appear? |
|  | **N/A - athenaNet creates free-standing claims and does not attempt to match the charges to an appointment.** |

* + 1. Processing Logic for Charge Messages
       1. Charges with Unmatched Appointments

There are two standard options for how athenaNet can handle DFT messages that fail to match to an appointment. Of course, it’s possible that your use case is for billing free-standing claims with charges that don’t have an appointment in athena. Example use cases for free-standing claims would be billing ancillary charges such as for an in-house lab, or billing hospital professional fees. In the event that you *are* billing for encounters with an appointment in athenaNet, this scenario could arise if for example the sending system sends a null or incorrect appointment ID, or if the interface is relying on an external appointment ID which was somehow deleted from the appointment custom field in athenaNet. Here are the options:

* Process the DFT, creating a free-standing claim not associated to an appointment, and therefore leaving the appointment, if it exists, on the Missing Slips worklist.
* Do not process the DFT and instead place the message in an ERROR status in your Interface Message Queue Manager (IMQM). All messages in an ERROR status in the IMQM require practice review.

Consider your use case and select whether you want the interface to be able to create free-standing claims.

**Does your organization want athenaNet to create free-standing claims instead of routing the inbound charge message to the ERROR queue?**  athenahealth recommends selecting “Yes” only if your organization’s use case involves billing free-standing claims with charges that don’t have an appointment in athenaNet.

DID YOU KNOW? athenaNet only processes charge data from inbound P03 charge messages. athenaNet discards all other data, including demographic updates.

* + - 1. Deriving Required Claim Data

If your use case is for free-standing claims, athenaNet derives all data needed for building the claim from the DFT message and patient record. If appointments are part of the use case, athenaNet can derive some claim data from the matched appointment – specifically provider, location, and insurance. The alternative is pulling all data from the DFT message and patient record, even when matching to an appointment and removing it from the Missing Slips worklist.

In Table 11 below, select how athenaNet should build the claim.

Table 11 – Matching Appointment Derived Data

|  |  |
| --- | --- |
|  | Option |
|  | **Derive all applicable data from the appointment when matched.**  This includes the rendering provider, service location (department), and insurance. The supervising provider is pulled from your provider table as supervising provider defined for the rendering. If this relationship isn’t defined, supervising defaults to the rendering. |
|  | **Derive insurance from appointment and derive provider and department data from the DFT charge message.**  From which field in the charge message will athenaNet pull the service location (department)?  Select how athenaNet will derive provider data from the charge message:  Rendering Provider in FT1.20 and Supervising Provider from Rendering Provider (per relationship defined in your provider table, if that isn’t defined the supervising defaults to the rendering)  Rendering Provider in FT1.20 and Supervising Provider in FT1.21 |
|  | **N/A – Use case is for free-standing claims**  All data will be derived from charge message.  Select how athenaNet will derive provider data from the charge message:  Rendering Provider in FT1.20 and Supervising Provider from Rendering Provider (per relationship defined in your provider table, if that isn’t defined the supervising defaults to the rendering)  Rendering Provider in FT1.20 and Supervising Provider in FT1.21 |

* + - 1. Insurance Logic

If a charge matches to an appointment, the claim is created with insurance from the appointment. When a free-standing claim is created, insurance is pulled from the patient’s Quickview.

* + - 1. Charge Grouping

Some third-party vendor systems (especially ancillary systems like lab systems (LIS) and health information systems (HIS)) send separate charge messages for a single encounter with multiple charges. When this happens, athenaNet groups the charges onto the same unbilled claim. athenaNet uses these data elements to match the charge transactions to the claim:

* Patient information
* Service date
* Rendering provider
* Supervising provider
* Department
* Primary insurance
* Secondary insurance
  + - 1. Charge Combining

When athenaNet receives multiple charge messages for the same patient, procedure, and date, it overrides the original charge with the charge in the most recent charge message. It also updates the unit amount based on the most recent charge message, rather than combining the units from both charge messages.

1. Scope Approval

Please provide an **electronic** signature approving the scope of the interface outlined in this document.

I,      , agree to the interface design as described here in this document.

Date:

* 1. Additional Comments

Enter general interface comments and questions that the document or your athenahealth Interface Project Engineer didn’t address.

1. Appendices and Other References
   1. Interface Message Queue Manager

The Interface Message Queue Manager (IMQM) page in athenaNet is an interactive repository for all interface messages that pass through athenaNet. Use the IMQM to view messages or resolve common errors, such as missing providers, invalid procedure codes, or unknown departments. Review table 19 to understand how athenaNet defines each state. Messages in a final state (processed or deleted) remain in the queue for only 90 days.

Table 19 - Interface message processing state

|  |  |
| --- | --- |
| Processing state | Definition |
| SCHEDULED | Scheduled to be sent later |
| NEW | Placeholder for a new message and ready to be sent or received |
| DISTRIBUTED | Delivery or acknowledgement is pending for global interfaces |
| PENDING | Delivery or acknowledgement is pending |
| PROCESSED | Processed normally; remains in queue for only 90 days |
| ERROR | Generic error encountered; routed to client |
| CBOERROR | Billing related error encountered; routed to client |
| ATHENAERROR | Internal error encountered; routed to athenahealth Client Support Center |
| DELETED | Messages that have been deleted; remains in queue for only 90 days |

Table 20 lists each permission required to access and make changes to the IMQM. Your local system administer must grant the user permissions.

Table 20 - Interface Message Queue Manager permissions

|  |  |
| --- | --- |
| Permission | Use case |
| Interface Admin: View Message Queue | You want to view the IMQM. |
| Interface Admin: Map Insurance Messages | You need to map insurance messages. |
| Interface Admin: Map Messages (except Insurances) | You need to map all messages excluding insurance messages (e.g. provider and department mappings). |
| Interface Admin: File Upload Interface | You want to upload files via the interface. |

See the [Interface Message Queue Manager guide](http://www.athenahealth.com/developer-portal/developer-toolkit/support) for more information on the IMQM and your organization’s responsibility for resolving messages in ERROR and CBOERROR status.

* 1. Continuing Service and Support

Your interface is transitioned into our daily service and support structure within two weeks after go-live.

As a standard practice, athenahealth continuously monitors all client connections and notifies the specified contacts if an error occurs. athenaNet monitors all jobs and restarts them automatically if they’re idle. For details, see the [Interface Down Support document](http://www.athenahealth.com/developer-portal/developer-toolkit/support).

You can also access support in athenaNet directly if you have questions about or modifications to the interface: On the Main Menu, click **Support** and then click **Get Help.**