HealthFlow7

User Manual

Version 1.1.23

SquidSpark Inc.

All data shown in this manual is synthetic. No real patient data and no customer data appears in any screenshot or example.

Contents

About HealthFlow7 5

What this manual covers 5

Free, Trial and Professional 5

Installing HealthFlow7 6

What you need 6

Installing interactively 6

Installing silently 6

Starting for the first time 7

Concepts you need before you start 8

Workspaces 8

Field paths 8

Where your data is kept 8

Importing messages 9

Create a workspace 9

Import from a folder 9

Other ways in 9

Reading a message 11

Open a message 11

Read the fields 11

Edit and compare 12

Searching 13

Run a search 13

Operators 13

Saved searches 14

Profiling a whole feed 15

Discovering an interface specification 16

Comparing two feeds 17

Reading the result 17

Diagnosing a single message 19
Removing patient data 20
Producing interface documentation 22
Regression suites 23
Build a suite 23
Run it 23
Sending and receiving live HL7 25
Listen 25
Send 25
Resubmitting messages 27
Settings 29
Licence 29
Licence server 29
Database 30
MLLP defaults 30
AI provider 30
Appendix A - Search expression reference 31
Paths 31
Operators 31
Combining 31
Notes 31
Appendix B - Files and formats 32
Import 32
Export 32
Appendix C - Troubleshooting 33
A page says a feature is professional 33
Analytics or Spec Discovery numbers look stale 33
An import found nothing 33
The professional features locked themselves 33

Messages sent by Resubmit are not arriving 33
Appendix D - The data in this manual 34

About HealthFlow7

HealthFlow7 reads real HL7 v2 traffic and tells you what an interface actually sends, rather than what its written specification claims it sends. It is a desktop application: messages are imported into a database on your own machine and are never uploaded anywhere.

It was built for the people who keep interfaces running — integration engineers, interface analysts and healthcare IT consultants — and for four situations in particular: migrating a feed from one system to another and having to prove the new side matches the old; working out at three in the morning why one message failed when thousands succeeded; onboarding a trading partner whose documentation is thin or missing; and preparing real traffic for a test system with the patient data taken out.

What this manual covers

This manual describes HealthFlow7 version 1.1.23 on Windows. Each chapter is a task: what the screen is for, the steps to complete it, and what the result means.

> *Note Every message, patient name, MRN, address, telephone number and clinician name in this manual is invented. The screenshots were taken against a synthetic feed from a hospital that does not exist. No real patient data and no customer data appears anywhere in this document.*

> *Note HealthFlow7 is a developer and integration tool. It is not a medical device and is not intended for the diagnosis or treatment of patients.*

Free, Trial and Professional

Three modules are free forever, with no expiry and no account: the Viewer and editor, Advanced Search, and Workspaces and import. They are enough to open, read, validate and search HL7 traffic.

The professional modules are Analytics, Compare, Diagnosis, De-identify, Spec Discovery, Documentation, Regression, MLLP and Resubmit. A 14-day trial unlocks all of them, once per device. When the trial ends the application returns to the free tier — nothing is deleted, the professional pages simply lock again.

> *Note The trial is one per device, not one per person, and it is spent by buying as surely as by running it: once a device has activated a licence, deactivating it does not make the trial available again.*

Installing HealthFlow7

What you need

- Windows 10 version 2004 (build 19041) or later, or Windows 11. 64-bit.
- No prerequisites. The installer is self-contained — there is no .NET runtime or other framework to install first.
- The WebView2 runtime, which ships with Windows 11 and current Windows 10.
- Roughly 400 MB of disk space, plus room for the messages you import.

Installing interactively

- Run the MSI. Accept the licence agreement when the installer presents it.
- Choose whether you want a desktop shortcut. It is created by default.
- Finish the installer. Tick "Launch HealthFlow7" if you want it to start immediately.

HealthFlow7 installs for all users of the machine, so the installer asks for administrator approval.

Installing silently

For unattended deployment, run the MSI through msiexec from an elevated command prompt. A quiet install cannot raise a UAC prompt, so it will fail rather than prompt if it is not already elevated.

```
msiexec /i "HealthFlow7-x64.msi" /qn /norestart
```

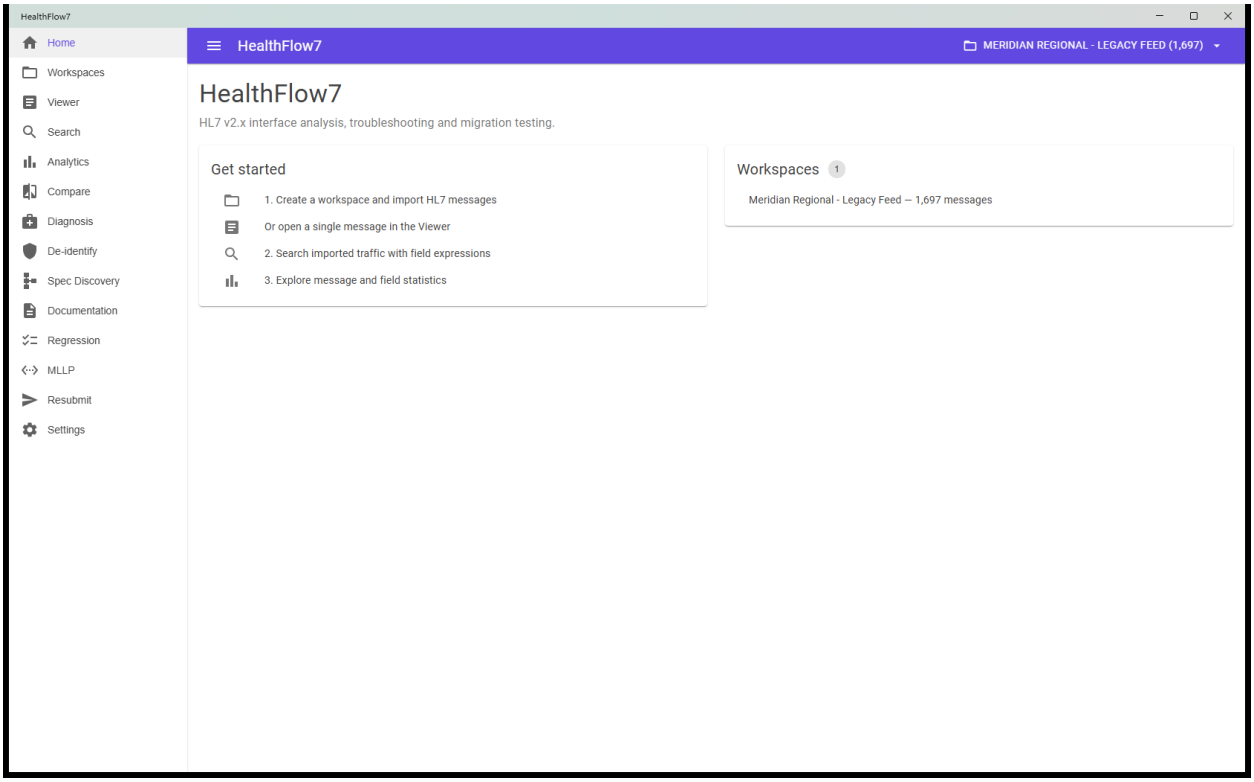
Useful variations:

```
msiexec /i "HealthFlow7-x64.msi" /qn INSTALLDESKTOPSHORTCUT=0

msiexec /i "HealthFlow7-x64.msi" /qn INSTALLFOLDER="D:\Apps\HealthFlow7"

msiexec /i "HealthFlow7-x64.msi" /qn /l*v "%TEMP%\hf7-install.log"
```

Note A silent install does not display the licence agreement, because that page belongs to the installer's user interface. When you deploy this way, your organisation is accepting the terms on behalf of its users.

Starting for the first time

On first run HealthFlow7 offers you the 14-day trial, an activation code box, and the option to continue on the free tier. You can start the trial later from Settings or from any locked page, so there is no penalty for choosing the free tier now.



The Home screen. The left-hand navigation lists every module; the panel on the right lists your workspaces and how many messages each holds.

Concepts you need before you start

Workspaces

A workspace is a named collection of messages — typically one interface, or one capture window from one interface. Almost every other page operates on the workspace that is currently active, which is named in the bar at the top right of the window.

Re-importing into a workspace appends to it rather than replacing it, so a feed can be topped up over several days. Statistics are recomputed across everything the workspace holds, not just the newest import.

Field paths

HealthFlow7 addresses any part of a message with a field path:

```
SEG-field.component.subcomponent      PID-5.1      MSH-9      OBX-3.2
```

A path with no component matches the whole field, and matching is by leading component — so MSH-9 = ADT^A01 also matches a message whose MSH-9 is ADT^A01^ADT_A01. Repeating fields match if any repetition matches. Comparisons know their data types, so dates compare as dates and numbers as numbers rather than as text.

Where your data is kept

Messages go into a local SQLite database file, and settings into a JSON file beside it, both under your own Windows user profile. Nothing is sent anywhere. The only outbound connections HealthFlow7 makes are licence checks, and — if you configure one yourself — an AI provider.

> *Note Because the database is per Windows user, two people signing in to the same PC each have their own workspaces and cannot see or disturb each other's imports.*

Importing messages

This is the first thing to do in a new installation: create a workspace and fill it.

Create a workspace

- Open Workspaces from the navigation.
- Type a name into "New workspace name". Name it after the interface and the capture, for example "Meridian Regional - Legacy Feed".
- Click CREATE. The new workspace becomes the active one.

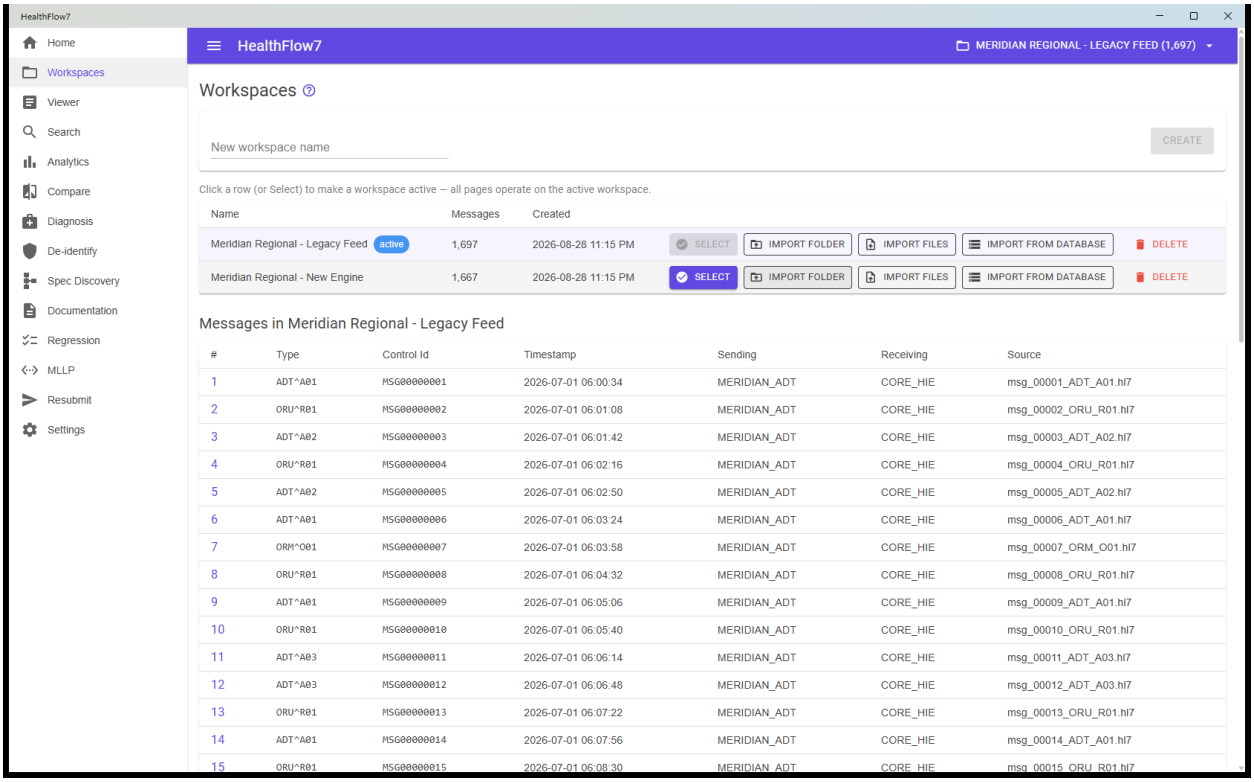
Import from a folder

- On the workspace's row, click IMPORT FOLDER.
- Choose the folder holding the messages. Sub-folders are included.
- Wait for the progress indicator in the title bar to finish. It reports the file count as it goes.

HealthFlow7 reads .hl7, .txt, .dat and .msg files. A file containing several messages is split automatically. Messages are ordered by file timestamp, so a folder capture replays in true arrival order.

Other ways in

- IMPORT FILES — pick individual files instead of a folder.
- IMPORT FROM DATABASE — run a SQL query against SQL Server, MySQL, PostgreSQL, Oracle, Microsoft Access, ODBC or OLE DB, and import the column holding the message text.



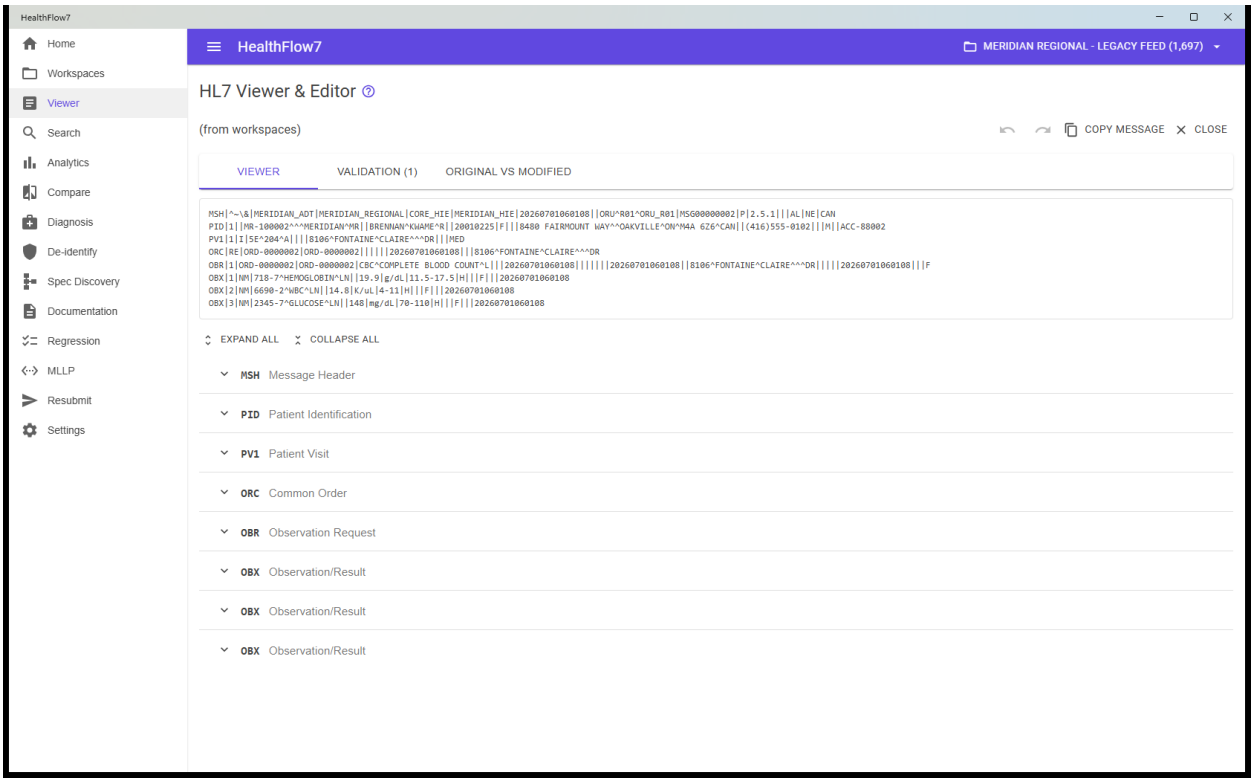
Two imported feeds. The message list beneath shows type, control id, timestamp, sending and receiving application, and the source file for every message in the active workspace.

Reading a message

The Viewer shows one message as raw text and as a segment-by-segment breakdown, with every field named from the HL7 dictionary.

Open a message

- From the Workspaces message list, click the number in the # column.
- Or open the Viewer directly and drag an HL7 file onto it.

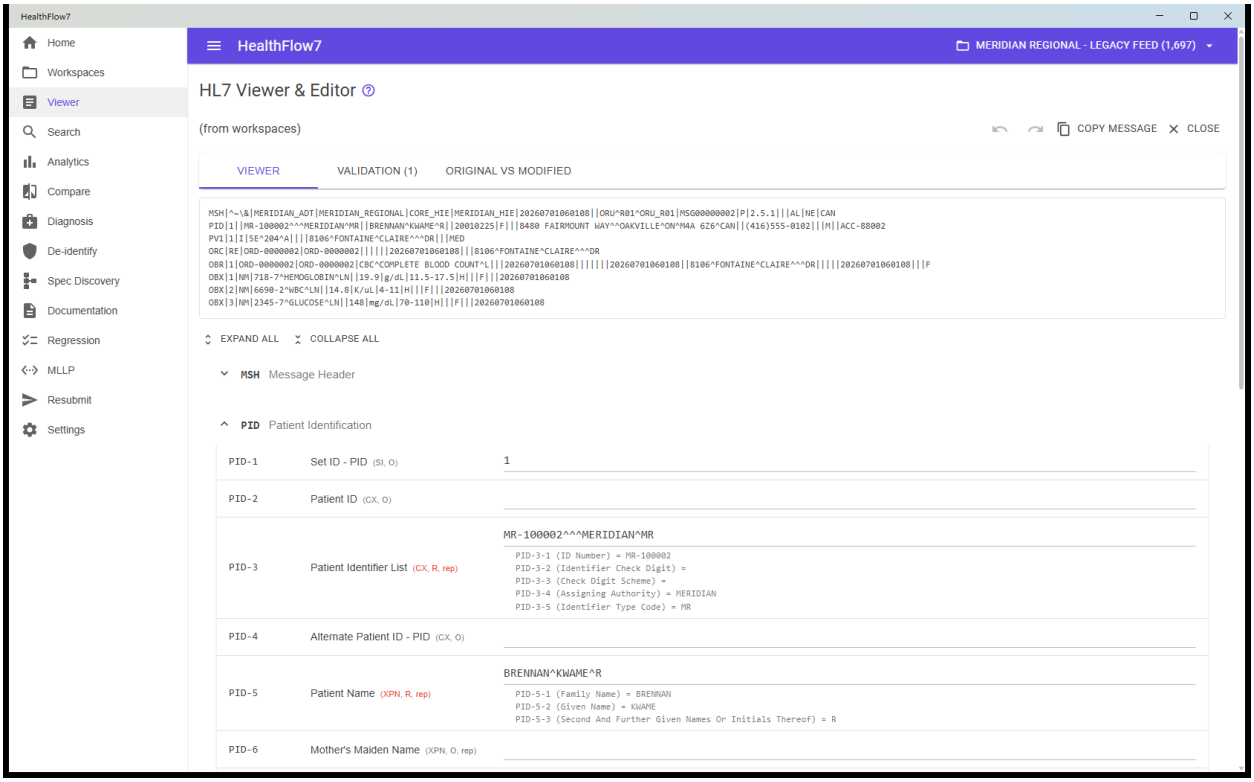


A lab result in the Viewer. The raw message is at the top; each segment below can be expanded. The VALIDATION tab counts structural and data-type problems.

Read the fields

- Click any segment header to expand it, or use EXPAND ALL.

Each field shows its position, its dictionary name, its data type and whether the standard marks it required, optional or repeating. Composite fields are broken down to component level with each component named.



PID expanded. PID-3 and PID-5 are decomposed to their components, each labelled from the HL7 dictionary.

Edit and compare

Any field can be edited in place. Validation re-runs as you type, and the ORIGINAL VS MODIFIED tab shows exactly what you changed. Undo and redo are available, and COPY MESSAGE puts the current text on the clipboard.

Searching

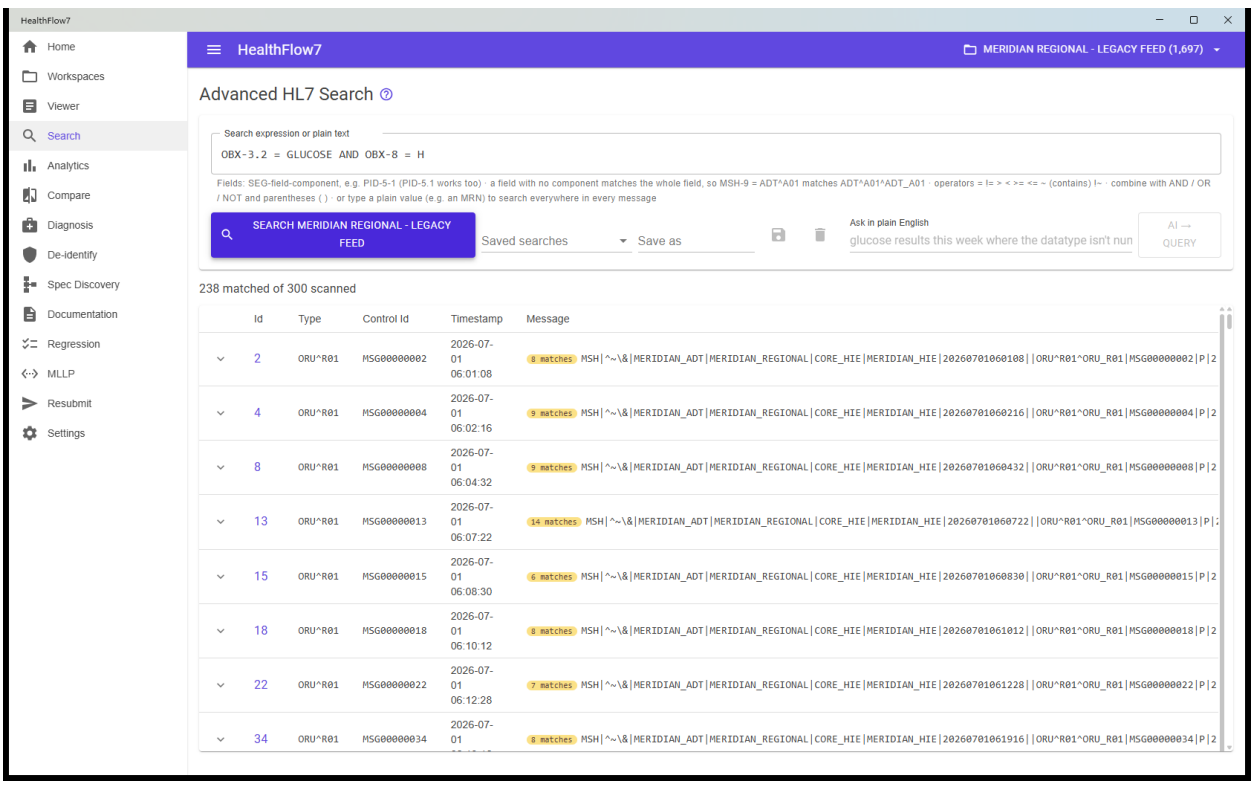
Search finds messages either by a plain value anywhere in the message, or by a precise field-path expression.

Run a search

- Open Search. It operates on the active workspace.
- Type an expression, or a plain value such as an MRN.
- Click SEARCH.

```
OBX-3.2 = GLUCOSE AND OBX-8 = H
```

Results list the matching messages with a count of matches inside each one, and every match is highlighted in the message text. Expand a row to see the whole message.



A field-path search for high glucose results, with per-message match counts.

Operators

- = and != — equality, matching by leading component.
- > < >= <= — ordered comparison, aware of dates and numbers.
- ~ — contains.

- !~ — does not contain.
- AND, OR, NOT and parentheses to combine them.

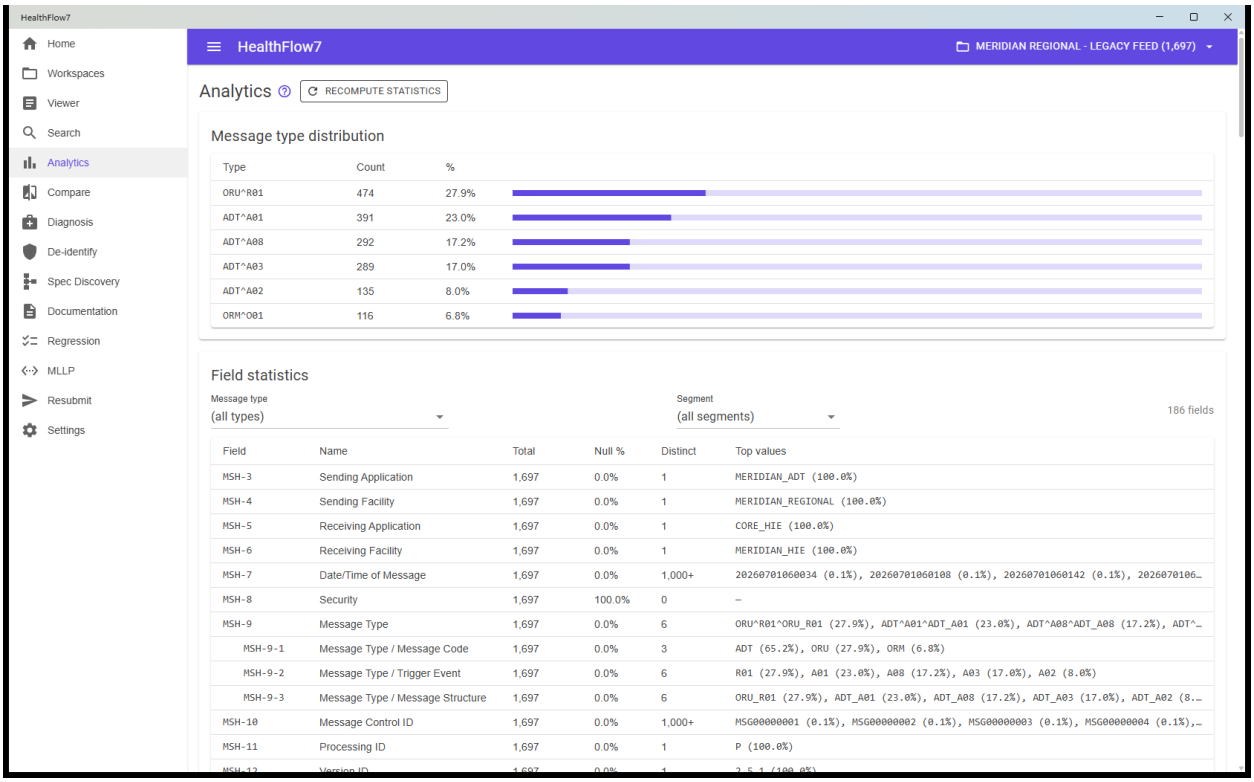
Saved searches

Give an expression a name in the "Save as" box and it appears in the Saved searches list for reuse. Saved searches are a good place to keep the handful of queries you run every time an interface misbehaves.

Profiling a whole feed

Analytics answers what a feed contains: which message types arrived, and for every field, how often it is populated and what values it carries.

- Open Analytics. Statistics are computed at import, so the page fills immediately.
- Use the Message type and Segment selectors to narrow the field list.
- If the numbers look stale after an import, click RECOMPUTE STATISTICS.



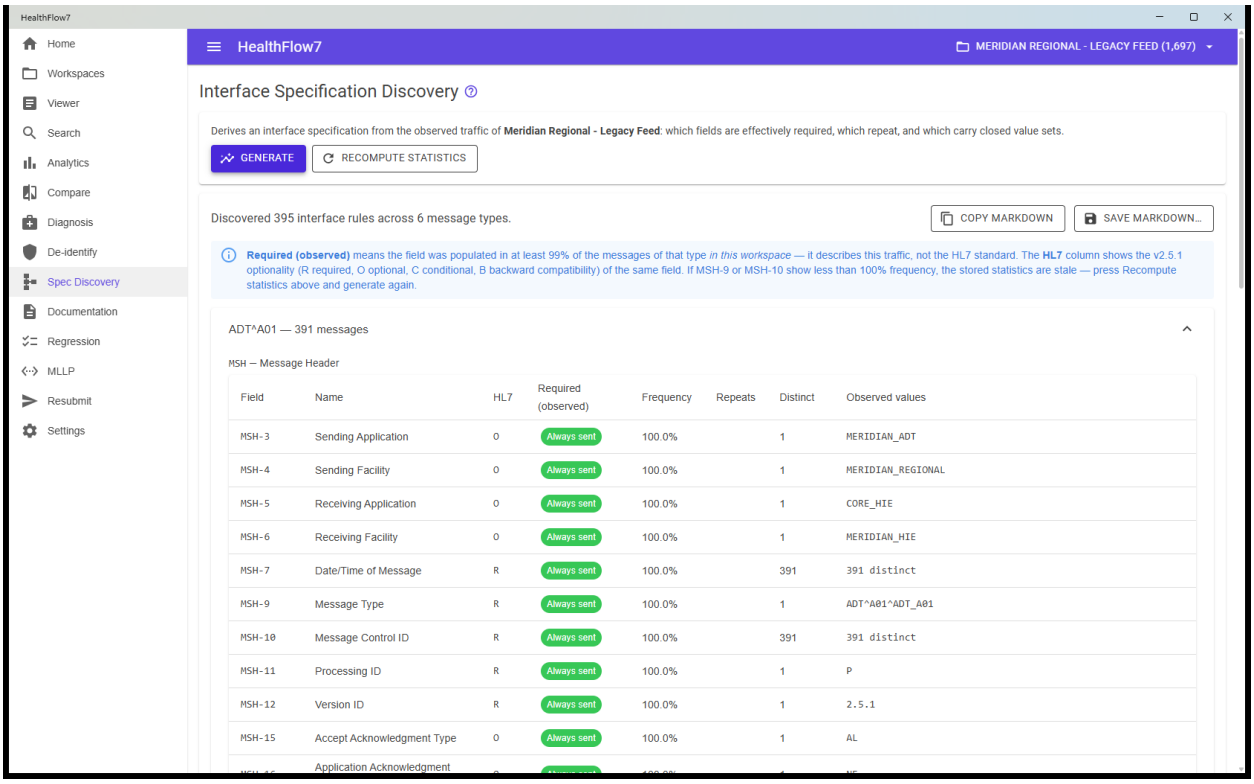
Message type distribution across 1,697 messages, and the first of 186 profiled fields with null percentage, distinct value count and top values.

The column to read carefully is Top values. A short list with high coverage means you have found a code set — a field that looks free-text in the specification but in practice only ever carries six values.

Discovering an interface specification

Spec Discovery reverse-engineers a specification from traffic you have already captured. It is the answer when a trading partner's documentation does not exist or cannot be trusted.

- Open Spec Discovery and click GENERATE.
- Read the rules per message type. Each field shows how often it was populated, whether it repeats, and what values were observed.
- Click COPY MARKDOWN or SAVE MARKDOWN to take the specification away.



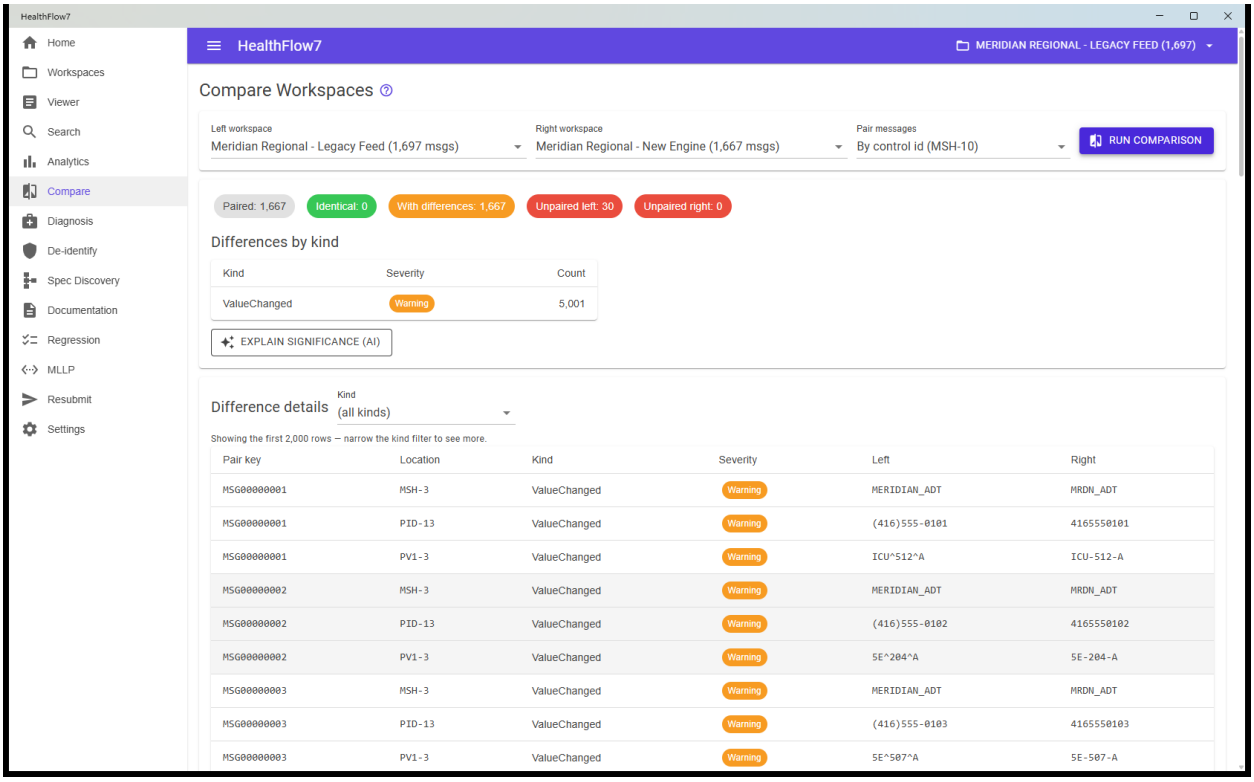
395 interface rules discovered across six message types.

> *Note The "Required (observed)" column is statistical — it means the field was populated in at least 99% of that type's messages in this workspace. The HL7 column beside it is the standard's own optionality. They are independent, and the gap between them is usually the interesting part.*

Comparing two feeds

Compare is the core check for a migration: it pairs the messages of two workspaces and reports every field that differs.

- Import the old interface into one workspace and the new one into another.
- Open Compare and choose the two workspaces.
- Choose how to pair them — by message control id (MSH-10), or by position if the new system renumbers.
- Click RUN COMPARISON.



A migration comparison. 1,667 pairs, 30 messages unpaired on the left, and 5,001 field differences — with the three real defects visible in the detail rows.

Reading the result

– Paired, Identical and With differences count the messages that matched up.

– Unpaired left and Unpaired right are the important ones — they are messages the other side never sent. In the example above, 30 unpaired on the left is a run of messages the new engine silently dropped.

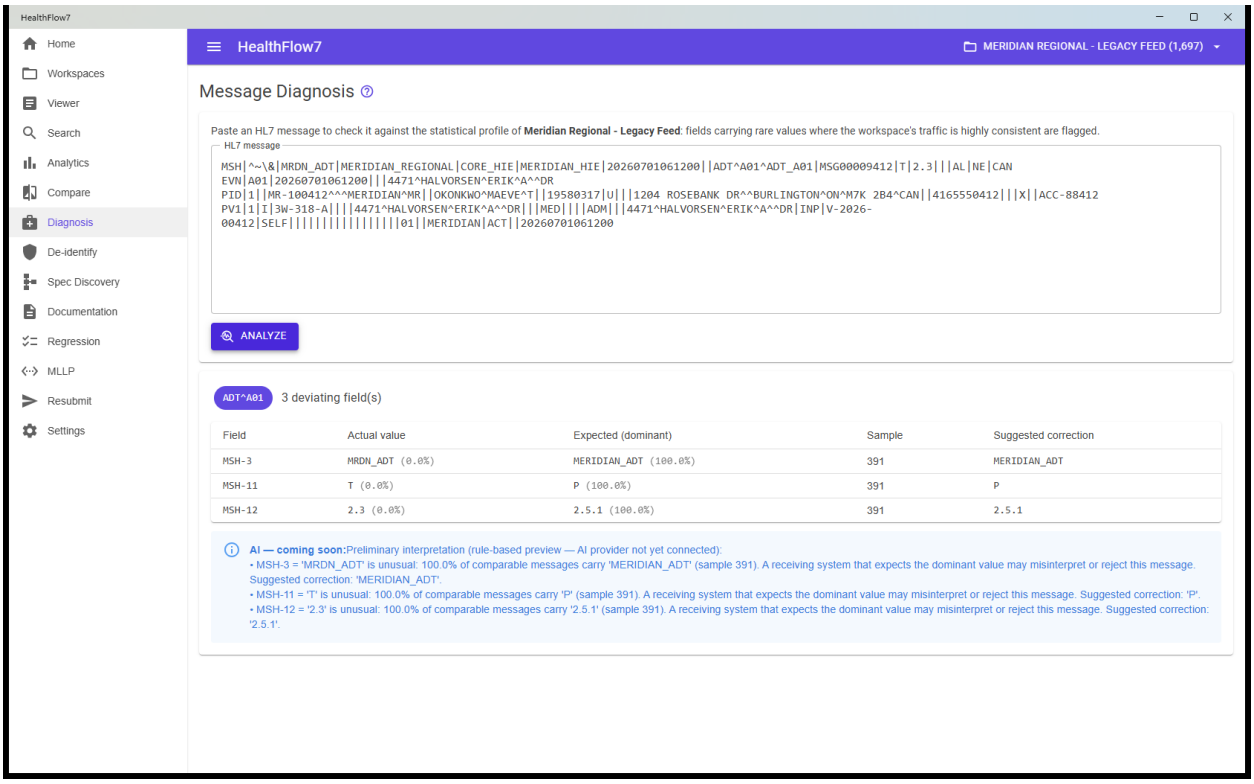
– Differences are graded by severity, and the detail table names the field, the left value and the right value.

In the screenshot the new engine renamed its sending application (MSH-3), reformatted the point-of-care code (PV1-3) and stripped the punctuation from telephone numbers (PID-13). All three are the kind of change that passes a smoke test and breaks a downstream system.

Diagnosing a single message

Diagnosis checks one suspect message against the statistical profile of a healthy workspace and names the fields carrying an unusual value.

- Open Diagnosis. It compares against the active workspace, so make sure that workspace holds traffic you consider good.
- Paste the message that failed.
- Click ANALYZE.



A message flagged on three fields, each with the value it carries, the value the feed almost always uses, and a suggested correction.

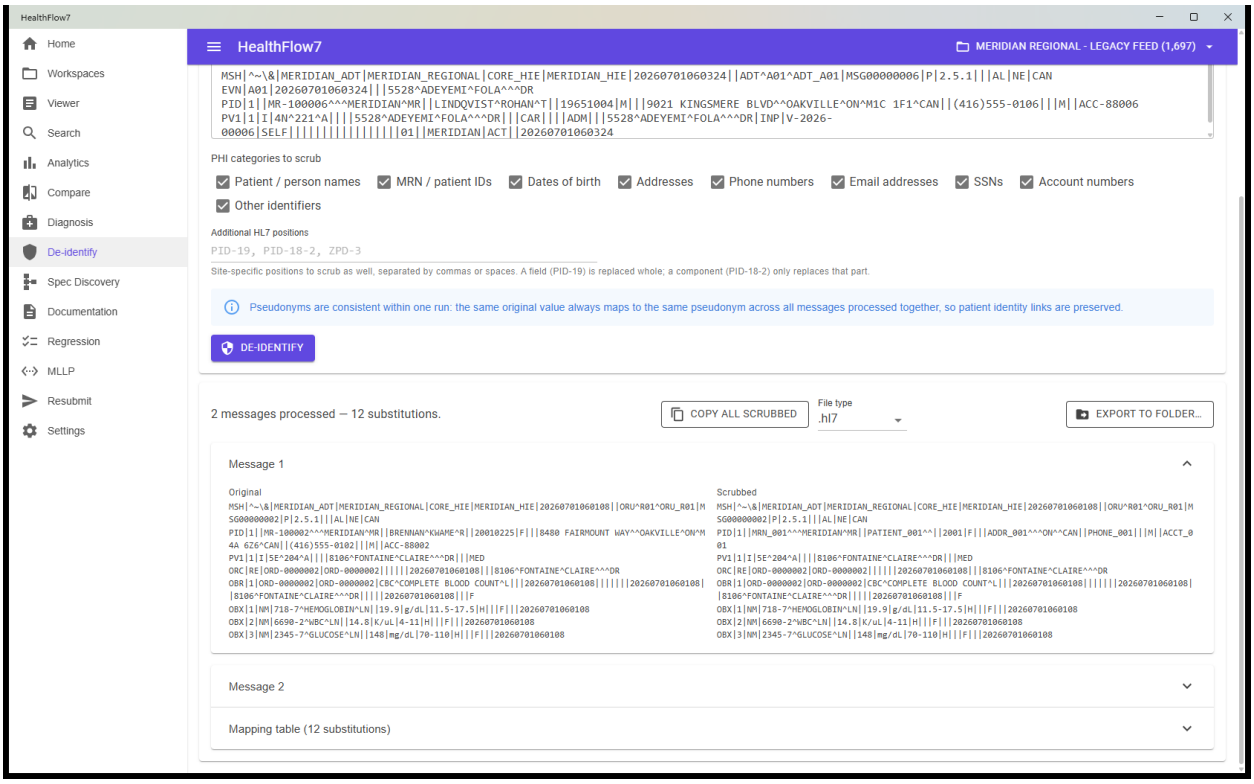
The thresholds are deliberately strict, so names, timestamps and control identifiers — which are different in every message by design — stay out of the results.

> *Note The narrative panel is marked "AI - coming soon". It is a rule-based preview; no AI provider is bundled with this release. Every statistic beside it is real.*

Removing patient data

De-identify replaces protected health information with consistent pseudonyms, so messages can be shared with a vendor or loaded into a test system.

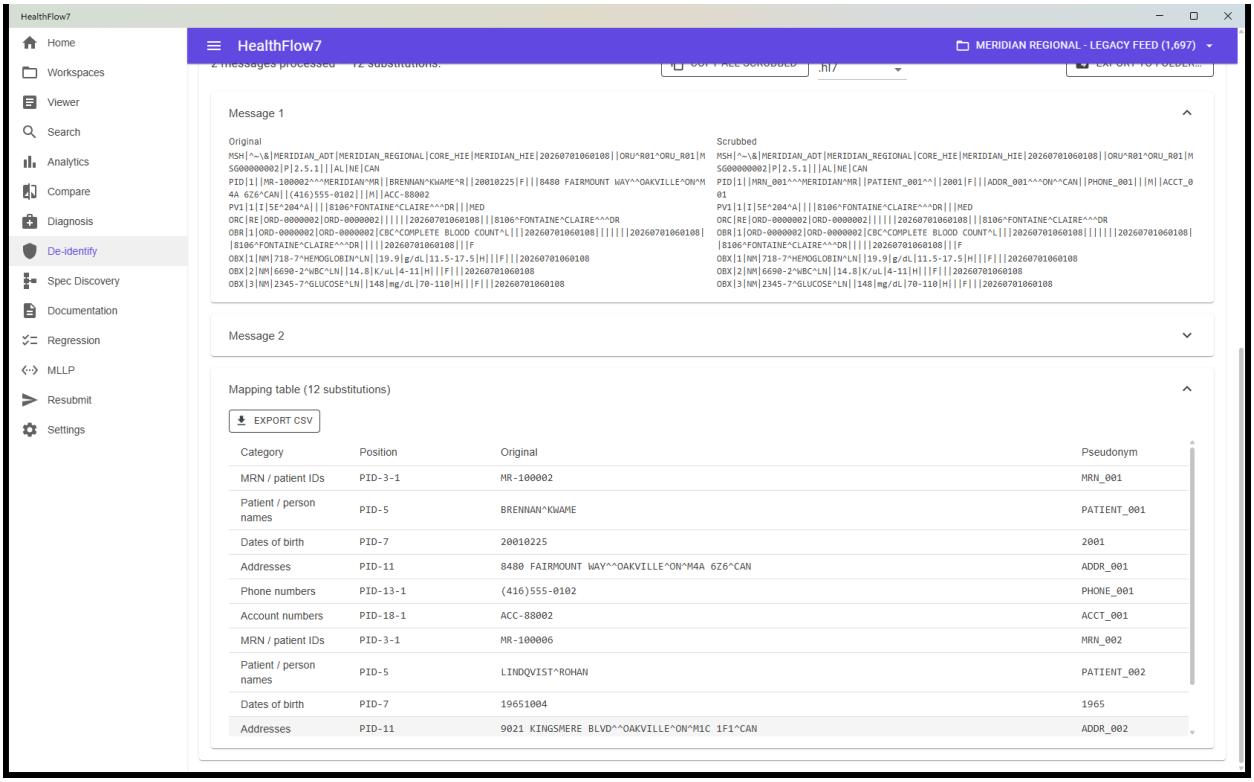
- Open De-identify and choose a source: paste text, pick files, or the whole active workspace.
- Tick the categories to scrub. All are on by default.
- Add any site-specific positions, such as a Z-segment field, in "Additional HL7 positions".
- Click DE-IDENTIFY.



Original and scrubbed side by side. Structure, assigning authorities and codes survive intact, so the message still parses and routes.

The same original value always becomes the same pseudonym within one run, so a patient appearing in 500 messages remains one patient and the messages still link together.

- Open the mapping table to see every substitution, and EXPORT CSV if you need to be able to re-identify later.



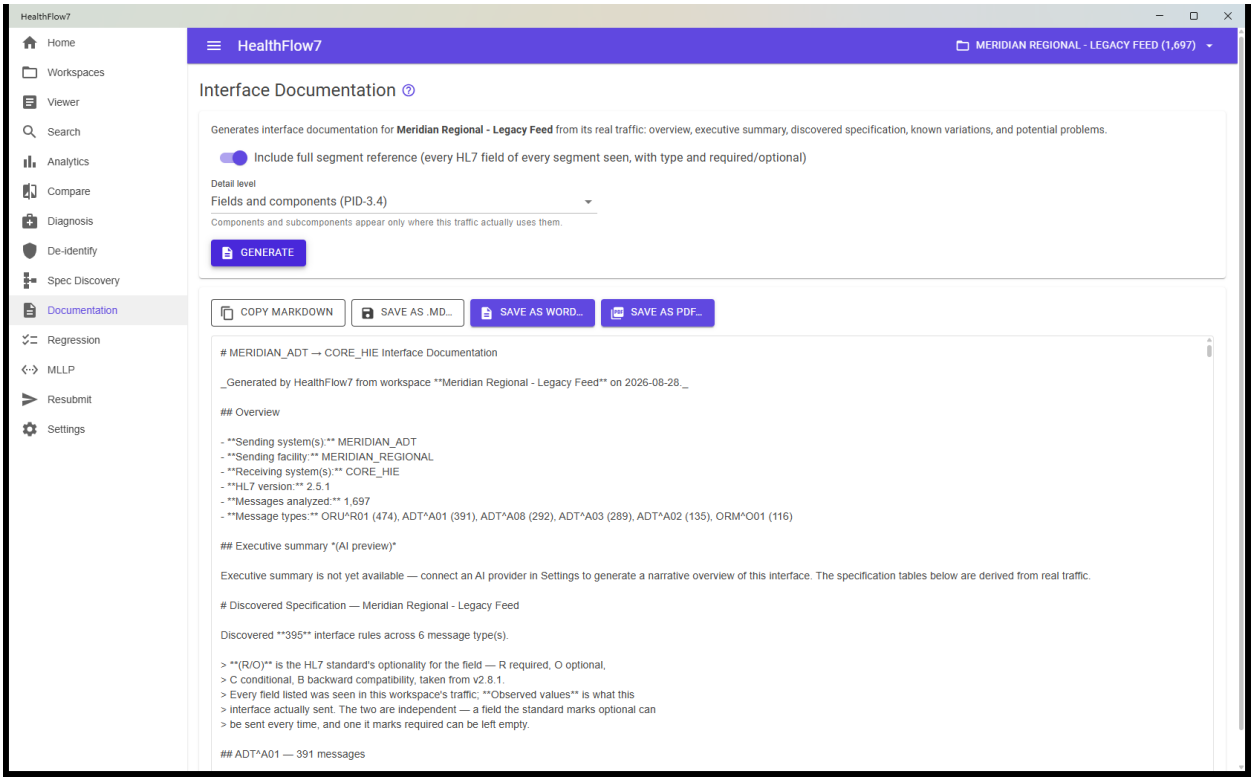
The substitution map: category, HL7 position, original value and the pseudonym that replaced it.

> *Note The mapping table is the only thing that can reverse the process. Treat it as sensitive — it is the link back to real patients.*

Producing interface documentation

Documentation generates a written document for a feed from its real traffic: an overview, the discovered specification, known variations and a full segment reference.

- Open Documentation.
- Choose the detail level, and whether to include the full segment reference.
- Click GENERATE.
- Export with SAVE AS WORD, SAVE AS PDF or SAVE AS .MD, or COPY MARKDOWN.



A generated interface document, ready to export to Word, PDF or Markdown.

The Word output uses real headings and tables, so it can be branded and sent to a trading partner without reformatting.

Regression suites

A regression suite is a set of saved messages, each with an expectation about it. Re-run the suite after any interface change to prove that nothing quietly broke.

Build a suite

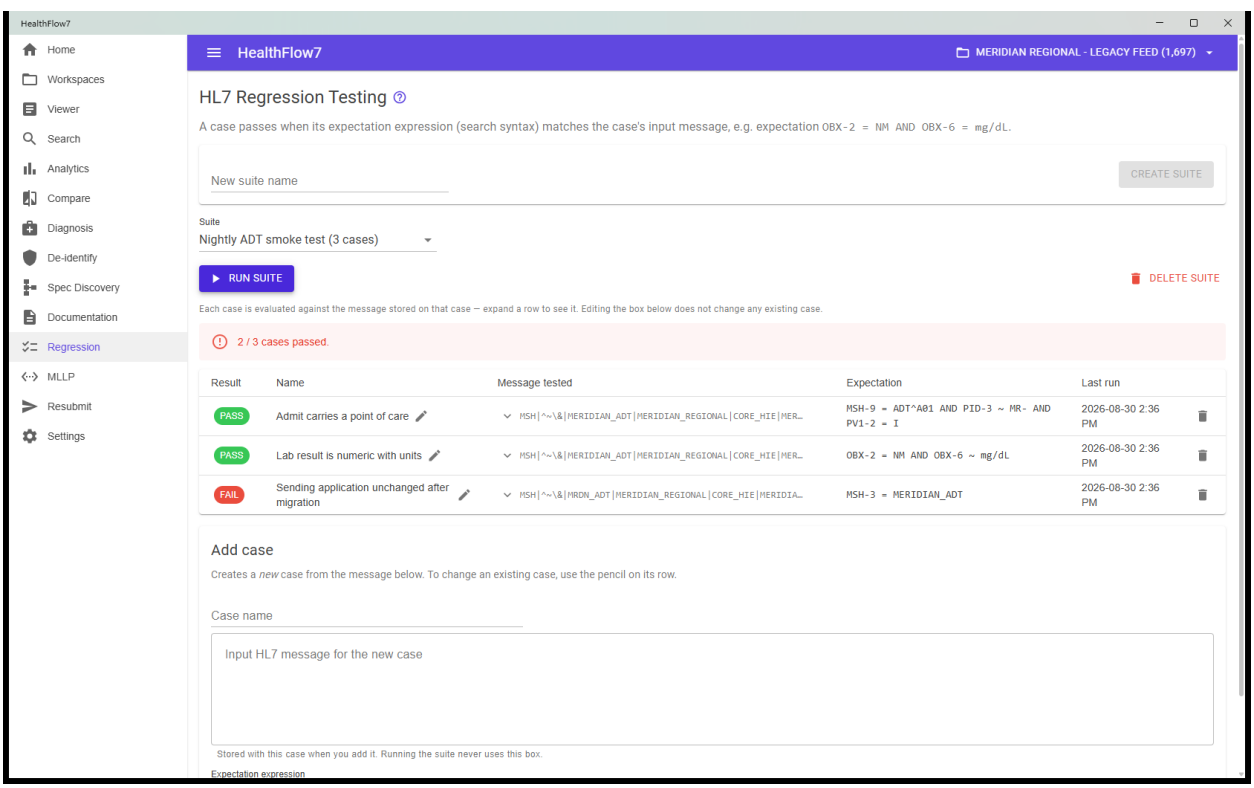
- Open Regression, type a suite name and click CREATE SUITE.
- In "Add case", give the case a name and paste the message it should be checked against.
- Write the expectation using the same expression syntax as Search.
- Click ADD CASE. Repeat for each case.

```
MSH-9 = ADT^A01 AND PID-3 ~ MR- AND PV1-2 = I
```

A case passes when its expectation matches the message stored on that case. The message is stored with the case, so a suite keeps working no matter what happens to your workspaces.

Run it

- Click RUN SUITE. Every case is evaluated and the pass count is reported.



Three cases, two passing and one failing. The failure is the point: the message came from the migrated engine, which no longer sends MERIDIAN_ADT in MSH-3.

The best cases to keep are the ones that already went wrong in production. Capture the message that broke, write the expectation that it should have satisfied, and that particular failure can never come back unnoticed.

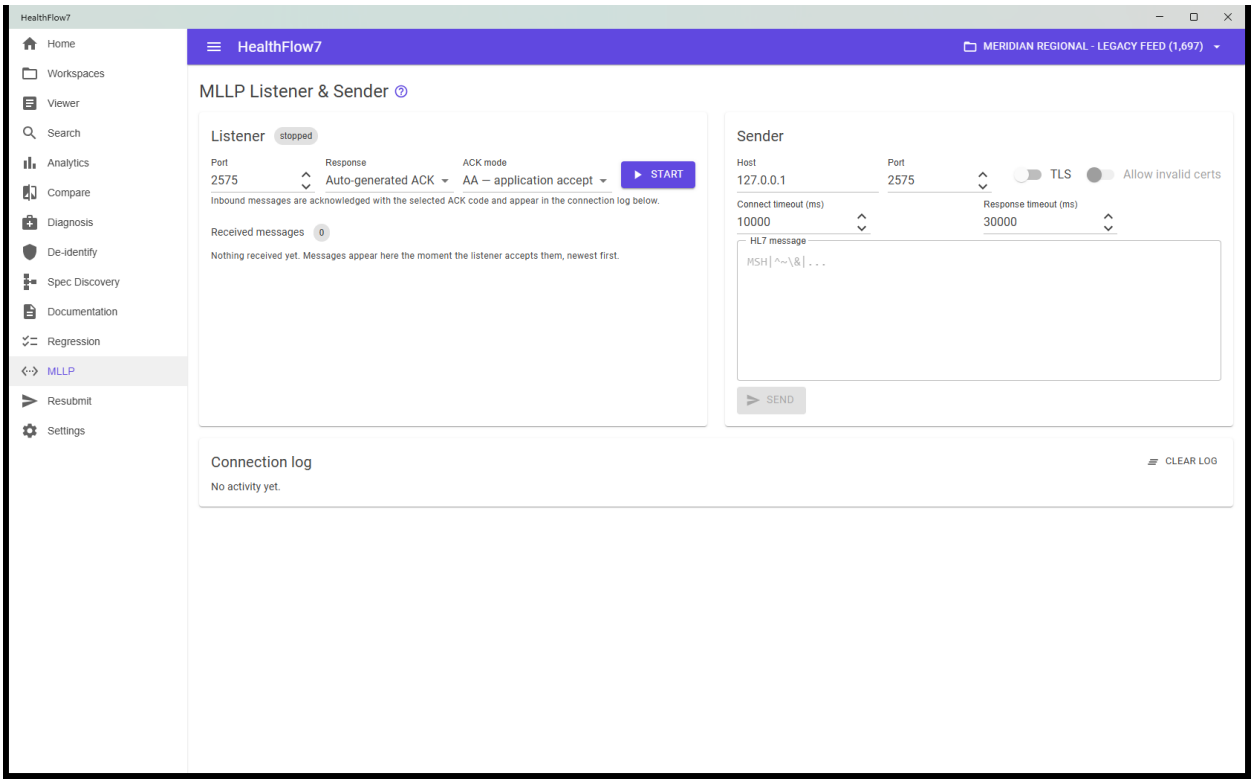
> *Note Running a suite transmits nothing. It is always safe to run.*

Sending and receiving live HL7

The MLLP page runs a listener that accepts messages and answers them, and a sender that transmits one message and shows you the acknowledgement.

Listen

- Set the port and choose the response mode.
- Choose the ACK code to answer with — AA, AE or AR, and CA, CE or CR.
- Click START. Received messages appear immediately, newest first.



The MLLP page. Listener on the left, sender on the right, and a connection log beneath recording both halves of every exchange.

Send

- Enter the host and port of the system you are testing.
- Turn on TLS if the far end requires it.
- Paste the message and click SEND.

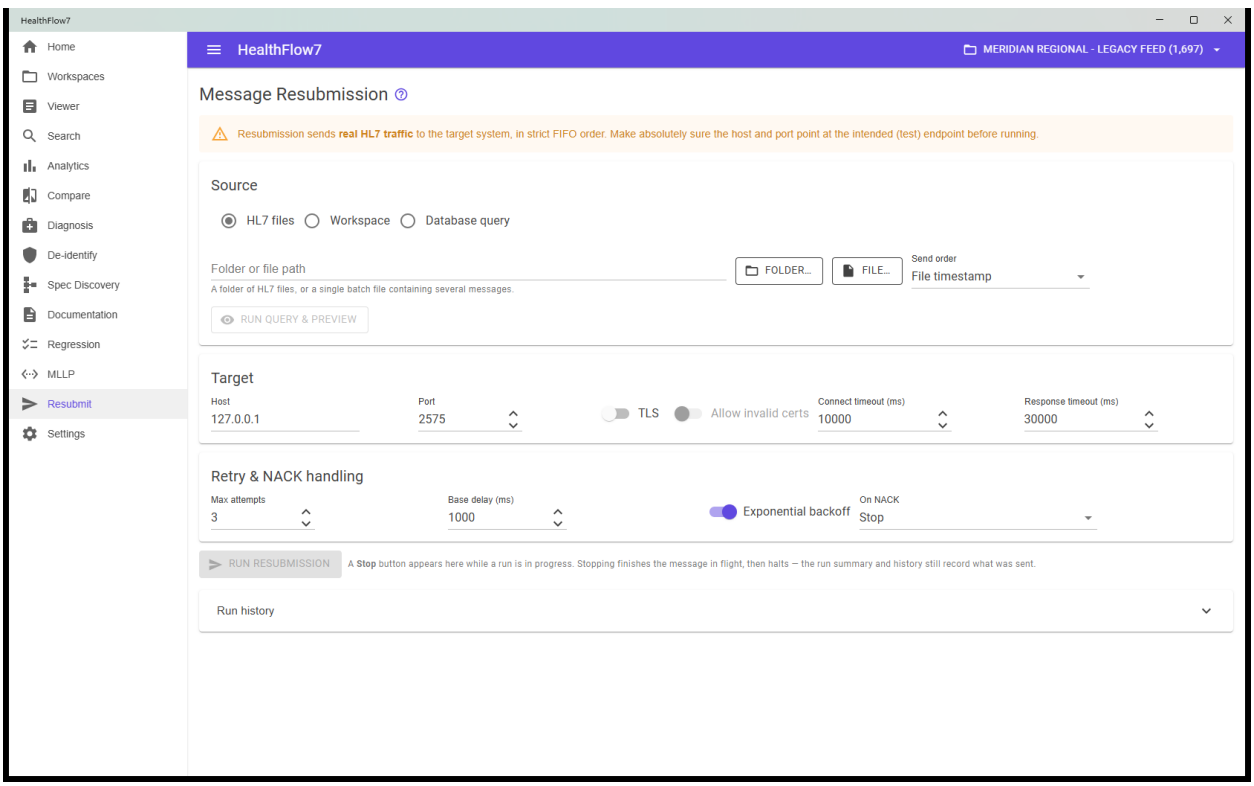
The acknowledgement is parsed and displayed with the round-trip time. Fixed-response mode on the listener is useful for reproducing a malformed or unusual ACK that a downstream system mishandles.

Resubmitting messages

Resubmit replays messages to an MLLP endpoint in strict FIFO order: message N+1 is never sent until N has been acknowledged or has conclusively failed.

> *Note Resubmission sends real HL7 traffic to the system you name. Check the host and port twice before running it.*

- Choose a source — a folder of files, the active workspace, or a database query.
- Set the send order, then preview it. Nothing is transmitted until you run the batch.
- Set the target host and port, and TLS if required.
- Choose the retry and NACK policy: how many attempts, the delay between them, and whether to stop, skip or retry when a message is not acknowledged.
- Click RUN RESUBMISSION.

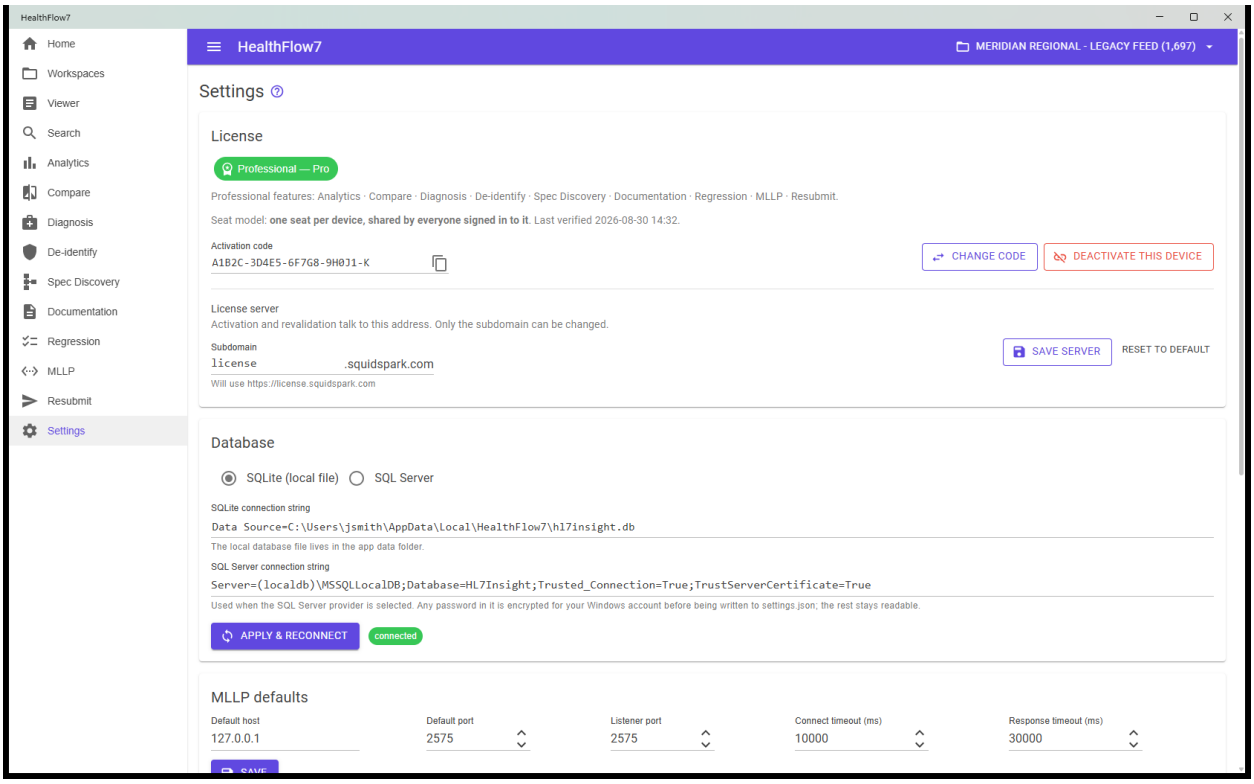


Resubmission configured but not yet running. Source, target and NACK policy are all set before anything is sent.

When the run finishes you get a summary: duration, throughput, and a row for every message that did not acknowledge. Runs are kept in Run history.

> *Note Run history records what was sent and when, but never the message body or the raw acknowledgement — that content stays in the source system.*

Settings



The Settings page. Licence at the top, then storage, MLLP defaults and the AI provider.

Licence

Shows your tier, the professional features it covers, and — for an activated licence — the seat model and when it was last verified with the licence server. Enter an activation code here, change it, or release the licence from this device so it can be used on another.

An activated licence re-checks periodically. If HealthFlow7 cannot reach the licence server for an extended period it warns you, and eventually locks the professional features until it can check in again. Being offline briefly changes nothing.

Licence server

You can point HealthFlow7 at a different licence server by changing the subdomain. The rest of the address is fixed. Click SAVE SERVER to apply it, or RESET TO DEFAULT to go back.

Database

SQLite is the default and needs nothing installed — it is a single local file. Choose SQL Server for shared or very large workspaces; the schema is created automatically. Click APPLY & RECONNECT after changing either. Any password in a SQL Server connection string is encrypted for your Windows account before it is written to disk.

MLLP defaults

The host, ports and timeouts used to pre-fill the MLLP and Resubmit pages.

AI provider

AI-assisted narratives are not part of this release. The setting is present so a provider can be configured when they ship; until then those features return a clearly marked preview, and every other part of the product works without them.

Appendix A - Search expression reference

Paths

```
PID-5           whole field

PID-5.1         first component

PID-5.1.2       subcomponent

MSH-9           matches ADT^A01 and ADT^A01^ADT_A01
```

Operators

```
=    equals (leading-component match)

!=   does not equal

>    greater than           <    less than

>=   at least               <=   at most

~    contains               !~   does not contain
```

Combining

```
MSH-9-1 = ORU AND (OBX-2 = ST OR OBX-2 = TX) AND MSH-7 > 2026-08-01

NOT (PID-8 = M) AND PV1-2 = I
```

Notes

- Matching is case-insensitive.
- A repeating field matches if any repetition matches.
- Dates and numbers are compared as dates and numbers, not as text.
- Typing a plain value with no operator searches the whole message text.

Appendix B - Files and formats

Import

- File types: .hl7, .txt, .dat and .msg.
- A file may hold one message or many; multi-message files are split automatically.
- Folders are read recursively, oldest first by file timestamp.

Export

- Interface documentation: Word (.docx), PDF, Markdown.
- Discovered specification: Markdown.
- De-identification mapping: CSV.
- Scrubbed messages: HL7 files.

Appendix C - Troubleshooting

A page says a feature is professional

That module needs the trial or a licence. Start the trial from the page itself, or enter an activation code in Settings.

Analytics or Spec Discovery numbers look stale

Click RECOMPUTE STATISTICS. Statistics are written at import; recomputing rebuilds them over everything the workspace now holds.

An import found nothing

Check the file extensions. Only .hl7, .txt, .dat and .msg are read, and the folder is searched including sub-folders.

The professional features locked themselves

The licence has not been verified for too long. Connect to the network once and restart; Settings shows when it was last verified.

Messages sent by Resubmit are not arriving

Confirm the host and port, and whether the far end expects TLS. The connection log on the MLLP page records both halves of the exchange and is usually the fastest way to see what the other system replied.

Appendix D - The data in this manual

Every screenshot in this manual was taken against a synthetic feed representing an invented hospital, "Meridian Regional", sending ADT, ORU and ORM traffic to an invented health information exchange.

The scenario is a system migration. A legacy interface of 1,697 messages is compared with its replacement of 1,667, where the new engine renamed its sending application, reformatted point-of-care codes and telephone numbers, and silently dropped a run of 30 messages. That is what the Compare and Regression chapters are showing, and why the numbers in them are consistent from chapter to chapter.

No real patient data and no customer data was used to produce this manual.