User Guide

Practical guidance for using Open Clinical History.

Manually Importing a Patient History

Open Clinical History provides a browser-based workflow for manually importing an existing patient history from a plain-text document.

The manual import page is:

```
/history_import.php
```

It is designed for importing raw, unstructured clinical history without requiring the source information to have already been converted into clinical events, SNOMED CT concepts or anatomical mappings.

The workflow has four stages:

```
1 · Upload
      |
      v
2 · Extract
      |
      v
3 · Audit & Import
      |
      v
4 · Complete
```

The original document is preserved while Open Clinical History builds a structured longitudinal clinical history from it.

Before Importing a Patient

The clinical processing environment should already have been prepared.

A normal installation sequence is:

```
Configure Open Clinical History
        |
        v
Import SNOMED CT
        |
        v
Build Image/SNOMED Database
        |
        v
Configure Gemini / LLM
        |
        v
Import Patient History
```

The manual upload itself can be performed even if the clinical-processing environment is not ready.

However, **extraction cannot start** until the required runtime data and LLM configuration are available.

Runtime Requirements

Before extraction begins, Open Clinical History verifies that the required clinical lookup data exists.

This includes:

```
snomed_health_history_lookup
snomed_health_history_term_lookup
body_layer
body_layer_lookup
body_layer_term
```

and the tables required by the patient-history processing pipeline.

It also verifies that the SNOMED and image lookup databases were built from the same SNOMED release.

If these checks fail, extraction is disabled and the page provides links to:

SNOMED Import

and:

Image/SNOMED Summary Build

This prevents LLM requests being consumed before deterministic SNOMED and anatomical mapping resources are available.

LLM Requirements

The configured LLM must also be available before extraction can start.

The current implementation uses Gemini.

The page verifies:

- LLM processing is enabled
- a Gemini API key is configured
- the configured model is available to the application
- the daily request/token budget has not prevented processing

The screen also shows current LLM usage information including:

```
model
requests today
tokens today
```

Configuration is managed under:

Admin → Configuration

Step 1: Upload

Open:

Import Patient History

or navigate directly to:

```
/history_import.php
```

The first screen asks for patient information and a history file.

Patient Record Number

Required

For example:

```
MRN-123456
```

or:

```
UR-84721
```

The record number is the primary identifier used by the import system to locate the patient.

Open Clinical History searches for:

```
patient_record_number = supplied record number

and

source_system = history_import
```

If a matching patient already exists, that patient is reused.

If no matching patient exists, a new patient is created.

Use a stable identifier

The same patient should always be imported using the same record number.

For example:

```
First import:
MRN-123456

Later import:
MRN-123456
```

Changing the identifier may create another patient instead of locating the existing one.

Important Limitation: Existing Patient Timelines

The current implementation can locate and reuse an existing patient.

However, it does **not yet perform full cross-document deduplication against events that have already been committed to that patient's longitudinal history**.

For example:

```
Document A
2019 - Myocardial infarction
        |
        v
Committed patient event



Document B
Past history: myocardial infarction in 2019
```

The extraction pipeline performs extensive duplicate and summary reconciliation **within the document being imported**, but it does not yet comprehensively compare the new document against every previously committed clinical event for that patient.

Therefore, care should currently be taken when importing overlapping histories for an existing patient.

This is particularly relevant when importing repeated copies of:

- complete medical histories
- problem lists
- discharge summaries
- specialist letters containing copied past history

Name

Optional

The patient's display name may be entered when creating a new patient.

For example:

```
Jane Example
```

The display name is stored as a patient attribute.

It is **not** used for patient matching.

If a matching patient record number already exists, entering a different name does not currently update that patient's stored details.

Date of Birth

Optional

Use the date selector provided by the browser.

Internally the value is submitted as:

```
YYYY-MM-DD
```

For example:

```
1965-11-24
```

Open Clinical History validates that:

- it is a real calendar date
- it is not in the future
- the year is not earlier than 1880

If the patient already exists, entering a different date of birth does not currently update that patient.

Sex at Birth

Available options are:

```
Female
Male
Intersex
Unknown
```

The value is also used when selecting the patient's anatomical image asset set.

Currently:

```
male   → male asset set


other values → female asset set
```

The anatomical catalogue available in the installation determines which artwork is ultimately available for visualisation.

History File

The manual importer currently accepts:

plain-text files only

Recommended extensions are:

```
.txt
.text
```

The browser file selector accepts:

```
text/plain
```

The document may contain completely unstructured clinical text.

It does **not** need to contain structured JSON, SNOMED codes or predefined event records.

For example, a source document may contain:

```
Patient: Example Patient

1998
Appendicectomy.

2007
Diagnosed with hypertension.

March 2016
Admitted following sudden onset left-sided weakness and dysarthria.
CT confirmed an acute cerebral infarction.

Current problems:
Hypertension
Residual left-sided weakness
```

Open Clinical History is responsible for interpreting the structure.

Maximum File Size

The maximum manual-upload size uses the configured:

```
api_max_upload_mb
```

setting.

The default is:

```
2 MB
```

Although the setting is named for the API, it is currently also used by the manual history importer.

The value can be changed under:

Admin → Configuration

A file larger than the configured limit is rejected before processing.

Character Encoding

Uploaded documents should preferably use:

```
UTF-8
```

If the uploaded text is not valid UTF-8, Open Clinical History currently attempts to convert it from:

```
ISO-8859-1
```

before storing the document.

What Happens to the Uploaded File?

Open Clinical History reads the uploaded text and stores its normalised contents in the database.

The document record includes information such as:

- patient
- source filename
- MIME type

- byte count
- character count
- SHA-256 document hash
- original text
- processing status

The import code does not deliberately copy the original uploaded file into a permanent web-accessible upload directory.

The clinical text itself is persisted in the `history_document` database record.

Exact Duplicate Documents

When a document is uploaded, Open Clinical History calculates a SHA-256 hash of its normalised text.

The document hash is associated with the patient.

If **exactly the same text** has already been uploaded for the same patient, Open Clinical History reuses the existing `history_document` rather than storing another copy.

For example:

```
Patient 42
   +
identical clinical text
        |
        v
existing history_document reused
```

This prevents accidental duplicate uploads of an unchanged source document.

This should not be confused with clinical-event deduplication.

A modified document produces a different document hash and is treated as a new source document.

Upload the Document

After entering the patient information and selecting the text file, click:

Upload and create patient

Despite the button wording, an existing patient with the same record number is reused rather than duplicated.

After upload, the page moves to:

```
2 · extract
```

The stored document is assigned an internal document ID.

For example:

```
document #42
```

Step 2: Extract

After uploading the document, Open Clinical History displays information including:

- document ID
- source filename
- character count
- document status
- patient record number
- LLM request count
- LLM token usage

If the required runtime services are ready, click:

Start extraction

Extraction Runs in the Background

The browser does not perform the extraction itself.

`history_import.php` launches:

```
history_job.php
```

using the server's CLI PHP installation.

Conceptually:

```
history_import.php
       |
       | start background process
       v
history_job.php
       |
       v
Clinical processing pipeline
```

The browser then polls a shared job-status file to display progress.

This means:

> It is safe to close the browser page after extraction has started.

The server-side extraction continues independently.

Manual Import and the Ingest Queue

The initial manual extraction does **not** currently use the normal `ingest_queue`.

Instead, `history_import.php` starts a dedicated background process for the document.

This differs from API imports, which normally enter the controlled ingest queue.

Conceptually:

```
Manual import

history_import.php
       |
       v
history_job.php



API import

history_api_import.php
       |
       v
ingest_queue
       |
       v
queue_worker.php
```

Only one extraction worker is allowed to process a particular document at a time.

Background Worker Requirements

Before starting extraction, Open Clinical History checks the CLI environment.

It verifies:

- a usable CLI PHP binary exists
- `pdo_mysql` is available to that PHP installation
- the PHP cURL extension is available
- `proc_open()` is available
- the shared runtime directory is writable

If one of these checks fails, the page displays:

The background worker cannot run

along with diagnostic information.

This is important because the PHP configuration used by the web server can differ from the PHP configuration used by CLI workers.

What Extraction Actually Does

The current patient-history pipeline consists of several separate processing stages.

```
Original text
      |
      v
1. Safety windows
      |
      v
```

```
2. Source segmentation
      |
      v
3. Atomic event extraction
      |
      v
4. Document-wide reconciliation
      |
      v
5. SNOMED resolution
      |
      v
6. Anatomical mapping
      |
      v
7. Clinical audit
      |
      v
Event proposals
```

The separation is deliberate.

The LLM is not simply asked to read the entire document and produce a final patient history in one operation.

---

1. Creating Safety Windows

PHP first divides the document into manageable overlapping safety windows.

The default target size is controlled by:

```
llm_max_chars_per_chunk
```

with a default of:

```
4500 characters
```

The overlap helps avoid losing context where an arbitrary processing boundary falls in the middle of a clinical section.

These windows are transport boundaries, **not clinical event boundaries**.

Maximum Windows Per Document

The number of safety windows is limited by:

```
llm_max_chunks_per_document
```

Default:

```
40
```

This setting is important.

If a document produces more windows than the configured maximum, excess windows are currently truncated.

Therefore, for unusually large patient histories, check the document status message for a truncation warning.

A document should not be considered fully processed if the source was truncated by this limit.

---

2. AI Source Segmentation

The first AI stage analyses each safety window and divides the source into coherent clinical blocks.

Examples include:

```
dated event
narrative event
```

```
procedure
investigation
allergy list
current problem list
past operation summary
copied discharge summary
family history
medication list
negative/rule-out statement
administrative text
```

The purpose is to recognise the structure of the source before trying to extract individual clinical events.

For example:

```
12 March 2018


Presented with sudden onset weakness of the left arm and dysarthria.
CT confirmed acute cerebral infarction.
```

should be understood as a coherent dated clinical section rather than as arbitrary pieces of text.

Every Source Span Is Preserved

The segmentation stage is designed to preserve source coverage.

If the AI fails to classify a piece of text, PHP still persists the uncovered portion as:

```
unclassified
```

rather than silently dropping it.

This gives later audit stages an opportunity to detect information that may not have been extracted correctly.

3. Atomic Clinical Event Extraction

A second AI stage extracts individual clinical events from the persisted source segments.

Events are deliberately atomic.

For example:

```
Left hemiparesis and dysarthria
```

may become:

```
Event 1: Left hemiparesis
Event 2: Dysarthria
```

Similarly:

```
Breast excision and sentinel node biopsy
```

may produce two independent procedure events when clinically appropriate.

The model extracts information including:

- event title
- concise clinical concept
- event type
- date information
- clinical status
- severity
- significance
- persistence
- laterality
- anatomical terms
- source evidence
- terminology search terms

At this stage the model does **not** invent SNOMED identifiers.

Non-Patient Events Are Identified

The extraction process explicitly distinguishes statements that should not become patient clinical events.

Examples include:

- negated conditions
- family history
- copied historical summaries
- administrative information
- medication lists where no independent clinical event is described

These may be retained for source understanding without becoming committed patient events.

---

4. Document-Wide Reconciliation

After individual source segments have been extracted, Open Clinical History considers the document as a whole.

This is important because clinical records frequently contain the same information in different places.

For example:

```
1987
Diagnosed with epilepsy.

...
```

```
Current problems
Epilepsy
```

The second statement should not necessarily create another diagnosis dated today.

The reconciliation stage considers:

- chronology
- repeated diagnoses
- copied past-history lists
- current problem lists
- status changes
- condition progression
- historical summaries

Its purpose is to construct a coherent set of canonical events from the entire source document.

---

5. SNOMED CT Resolution

After reconciliation, Open Clinical History attempts to resolve each clinical event against the local SNOMED database.

It uses the compact tables created by the SNOMED import:

```
snomed_health_history_lookup
snomed_health_history_term_lookup
```

Candidate concepts are generated locally.

Where an LLM is required to resolve ambiguity, it is allowed to choose only from the supplied local candidates.

Conceptually:

```
Clinical event
      |
      v
Local SNOMED search
      |
      v
```

```
Candidate concepts
      |
      v
Optional bounded AI choice
      |
      v
Verified local SNOMED concept
```

The model cannot simply invent a SNOMED concept identifier.

---

6. Anatomical Image Mapping

Once terminology has been resolved, the pipeline attempts to connect the clinical event to the installed anatomical image catalogue.

The principal runtime resources are:

```
body_layer_lookup
body_layer_term
```

The system considers:

- the resolved SNOMED concept
- anatomical site information
- laterality
- installed image candidates
- mapping safety rules

The result may be zero, one or several suitable image layers.

A valid clinical event does not necessarily require an anatomical image.

For example, some systemic or non-local conditions may have no useful body-layer representation.

---

7. Final Clinical Audit

Before the document becomes available for import, Open Clinical History performs several read-only audit stages.

These assess:

Source coverage

Did the extraction capture genuine clinical events present in each source segment?

Event fidelity

Are dates, laterality, status and clinical claims supported by the source?

Duplicate reconciliation

Are events that appear to describe the same clinical episode still duplicated?

SNOMED mapping

Were clinical events successfully resolved to appropriate terminology?

Timeline consistency

Do the events make sense when viewed across the complete chronology?

Body-layer suitability

Are the selected anatomical images appropriate to the clinical event?

Import Quality Score

The final audit produces an overall quality score from:

```
0 to 100
```

and a grade:

```
A
B
C
D
E
```

The score combines:

Component	Weight
Source coverage	30%
Event fidelity	25%
Duplicate reconciliation	15%
SNOMED mapping	15%
Date/status/laterality consistency	10%
Body-layer suitability	5%

The import screen also reports information such as:

- critical blockers
- warnings
- likely duplicates
- uncovered source events
- unresolved SNOMED events

The quality score is a processing-quality indicator.

It should not be interpreted as a measured percentage of clinical accuracy.

Live Extraction Progress

While the background worker is running, the page displays live progress.

Depending on the current phase, you may see activity such as:

```
Creating safety windows
AI source segmentation
AI atomic event extraction
Document-wide reconciliation
SNOMED candidate retrieval
SNOMED concept resolution
Body-layer mapping
Source coverage audit
Global consistency audit
```

The display also reports information such as:

- source windows
- source segments
- extracted events
- canonical events
- requests
- tokens
- individual section status

Extraction Can Take Time

Large clinical histories can require multiple LLM requests and significant local terminology processing.

Processing time depends on:

- document size
- number of source sections
- number of extracted events
- SNOMED ambiguity
- terminology repair requirements
- audit complexity
- LLM response latency

A large longitudinal patient history should not be expected to complete instantly.

Extraction Is Resumable

The pipeline persists its work as it progresses.

If the worker stops unexpectedly, the document may display:

The document says it is processing, but no active worker was found.

The page then provides:

Resume unfinished sections

Completed work is retained.

The resumed worker continues from the first unfinished processing stage rather than deliberately starting the complete AI extraction again.

This can preserve already completed LLM responses and reduce unnecessary model usage.

Step 3: Audit & Import

Once extraction and auditing have completed, the document enters:

```
proposed
```

status.

The screen displays the extracted proposals for review.

Each proposal can include information such as:

- event title
- date
- event type
- laterality
- severity
- significance
- clinical status
- persistence
- source type

- SNOMED concept
- terminology matching method
- anatomical layers
- mapping warnings
- audit status
- audit findings
- source excerpt

Source Evidence

Each extracted event retains a source excerpt.

For example:

```
"CT brain demonstrated an acute left middle cerebral artery infarction."
```

This allows the proposed structured event to be compared directly with the original evidence that produced it.

Source evidence is an important part of the audit trail.

SNOMED Matching Information

Where a SNOMED concept was resolved, the proposal shows:

- selected SNOMED term
- concept identifier
- matching method

Possible matching methods include:

```
exact SNOMED FSN
exact SNOMED synonym
```

```
grounded SNOMED selection
SNOMED prefix suggestion
no SNOMED match
```

The terminology result is generated from the local SNOMED catalogue.

Mapping Warnings

A proposal can also contain anatomical mapping warnings.

Examples may include:

- laterality conflicts
- inappropriate anatomical imagery
- insufficient anatomical evidence
- ambiguous body mapping

The current automatic-import stage takes these warnings seriously.

A proposal with mapping warnings is not automatically committed.

Run Auto-Import

When the proposals are ready, click:

Run Auto-Import

This does **not** blindly commit every extracted proposal.

Each proposed event passes a safety gate.

An event is automatically committed only when:

```
audit status = pass

AND
```

```
a SNOMED concept has been resolved

AND

the statement is not negated

AND

the statement is not family history

AND

the statement is not merely a copied historical summary

AND

there are no anatomical mapping warnings
```

Conceptually:

```
Extracted proposal
       |
       v
Audit passed?
       |
       +--- no ---> Learning queue
       |
      yes
       |
       v
SNOMED resolved?
       |
       +--- no ---> Learning queue
       |
      yes
       |
       v
Mapping safe?
       |
       +--- no ---> Learning queue
```

```
        |
       yes
        |
        v
Commit to patient history
```

Committed Events

Safe events are inserted into:

```
clinical_event
```

The committed record preserves information including:

- patient
- date and date precision
- title
- summary
- clinical status
- event type
- laterality
- severity
- significance
- terminology mapping
- source document provenance
- source evidence
- reviewer/import identity

Associated visual mappings are written to:

```
clinical_event_site
```

where suitable anatomical layers exist.

Events That Cannot Be Automatically Imported

A proposal that does not pass the automatic-import safety gate is marked:

```
rejected
```

from automatic import and routed to:

```
history_unmatched_queue
```

This does **not necessarily mean the clinical statement itself is wrong**.

It means the system did not consider the proposal safe enough for unattended commitment.

The item is categorised according to the problem.

SNOMED Concept Problems

An unresolved terminology item is classified as:

```
snomed_concept
```

For example:

```
Clinical event understood
        |
        v
No sufficiently reliable SNOMED concept
        |
        v
Learning queue
```

Body-Layer Problems

If SNOMED terminology was found but the anatomical mapping raised warnings, the item is classified as:

```
body_layer
```

Specificity Problems

If terminology exists but another ambiguity prevents safe commitment, the item may be classified as:

```
specificity
```

The Learning Queue

The learning queue provides a controlled mechanism for resolving terminology and mapping gaps.

This is deliberately separated from the original extraction.

The system can therefore improve its terminology knowledge without repeatedly asking the LLM to reinterpret the original patient document.

Apply Learned Mappings

If terminology for previously unresolved events is later resolved through the governed learning process, the import page can display:

Apply learned mappings

This operation:

- uses the newly resolved local terminology
- remaps the event against the current anatomy catalogue
- rechecks the original audit gate
- commits or refreshes eligible events
- does not repeat the original source extraction

The application stage itself does **not require another LLM extraction call**.

It is queued through the normal worker pool.

Governed Learning State

Where applicable, the page displays the learning state for the document.

This can include counts for:

```
Learning queue pending
Resolved terminology awaiting application
Queued
Running
Applied
Held for review
Failed
```

This allows unresolved clinical terminology to be managed independently from the original patient import.

Full Re-run Mapping

The screen also provides:

Full re-run mapping

This is an advanced repair operation.

It should **not** be used as the normal way to apply newly learned terminology.

A full mapping re-run:

1. removes proposed and rejected mappings for the document
2. resets mapping status
3. clears the document's unmatched queue entries
4. starts the processing worker again
5. rebuilds terminology and anatomical mapping

The previously persisted AI source extraction is retained, so the intent is to avoid repeating the original text extraction.

Use this when the mapping itself needs to be rebuilt, for example after a significant mapping-system correction.

For ordinary terminology learning, use:

Apply learned mappings

instead.

Step 4: Complete

After automatic import finishes, the document status becomes:

```
committed
```

The screen reports:

Import complete

and provides:

View Patient History

The committed clinical events can now be viewed as part of the patient's longitudinal history.

What `committed` Means

A committed **document** does not mean that every extracted proposal was committed.

For example:

```
24 proposals extracted


20 safe events committed


4 events routed to learning queue
```

The document can still be considered processed and committed.

The unmatched items remain available for governed learning and later application.

Patient History Output

Committed events become part of the patient's longitudinal history.

Conceptually:

```
Raw clinical document
        |
        v
Source segmentation
        |
        v
Atomic clinical events
        |
        v
Chronological reconciliation
        |
        v
SNOMED CT classification
        |
```

```
        v
Anatomical mapping
        |
        v
Clinical audit
        |
        v
Safe committed events
        |
        v
Longitudinal patient history
```

This is the central purpose of the manual history importer.

Recommended Import Workflow

For a new patient:

```
1. Open history_import.php
        |
        v
2. Enter stable record number
        |
        v
3. Enter optional demographics
        |
        v
4. Select plain-text history
        |
        v
5. Upload
        |
        v
6. Verify runtime checks
```

```
        |
        v
7. Start extraction
        |
        v
8. Allow background processing to finish
        |
        v
9. Review audit score and proposals
        |
        v
10. Run Auto-Import
        |
        v
11. Review learning-queue items if present
        |
        v
12. View Patient History
```

Troubleshooting

Start Extraction Is Disabled

Check the messages above the extraction controls.

Common causes include:

- Gemini is not configured
- LLM processing is disabled
- SNOMED import has not completed
- image/SNOMED database has not been built
- lookup tables are empty
- SNOMED and image databases were built from different releases
- the CLI worker environment is unavailable

Runtime Data Is Not Ready

Open Clinical History deliberately prevents extraction when deterministic terminology or image data is missing.

Run:

SNOMED Import

followed by:

Image/SNOMED Summary Build

and reload the patient import.

LLM Is Not Configured

Open:

Admin → Configuration

and verify:

- Enable LLM processing
- Gemini API key
- Gemini model
- request/token limits

A document may remain uploaded until the LLM configuration is corrected.

Background Worker Cannot Run

Check the diagnostics displayed on the page.

The CLI PHP environment requires:

```
PHP
pdo_mysql
curl
proc_open
writable shared run directory
```

The web PHP installation working correctly does not guarantee the CLI PHP installation has the same extensions.

Document Is Stuck on Processing

If the document reports `processing` but no active worker can be located, the page should offer:

Resume unfinished sections

Use this rather than uploading the source again.

Extraction Completed With No Events

The page explicitly warns when extraction finishes without producing proposals.

Do not treat this as a successful empty clinical history without investigating.

Review:

- the source document
- extraction progress
- worker log
- LLM configuration

A Valid Event Was Not Imported

Check the proposal's:

- audit status
- SNOMED match
- mapping warnings
- negation/family-history status
- learning-queue state

A clinically meaningful event may have been deliberately held because Open Clinical History could not establish a sufficiently safe structured representation.

Same File Was Uploaded Again

If the source text is identical and the same patient record number was used, Open Clinical History may return to the existing document because exact document-content deduplication is already implemented.

This is expected behaviour.

Current Limitations

The manual importer currently has several deliberate limitations.

Plain text only

The upload screen currently accepts plain-text clinical histories.

PDF, Word, image and scanned-document ingestion are not handled directly by this screen.

Source material must first be converted to text.

Existing patient demographics are not updated

A matching record number reuses the existing patient.

Name, date of birth and sex supplied during the later import do not currently update that patient.

Cross-document timeline deduplication is not yet complete

The pipeline performs sophisticated reconciliation within the source document being processed.

It does not yet fully reconcile a newly imported document against every clinical event already committed to the patient's existing longitudinal timeline.

This should be considered when importing overlapping source histories.

Related Components

Component	Purpose
`history_import.php`	Manual patient import and review interface
`history_job.php`	Background extraction worker
`lib/history.php`	Core patient-history ingestion and mapping services
`HistorySegmentationService`	Converts safety windows into coherent source segments
`HistoryExtractionService`	Extracts atomic clinical events
`HistoryReconciliationService`	Reconciles the complete source chronology
`ResolveTerminologyAndBodyStep`	Resolves SNOMED terminology and anatomical layers
`HistoryAuditService`	Performs final source and consistency audits
`HistoryLearningApply`	Applies governed terminology learning after extraction

Summary

The manual patient-history importer is designed to turn:

```
Raw unstructured clinical text
```

into:

```
A source-backed,
SNOMED-classified,
anatomically mapped,
audited longitudinal clinical history
```

without requiring the person performing the import to manually structure the source record first.

The process deliberately separates:

```
Source storage
    |
Interpretation
    |
Terminology resolution
    |
Anatomical mapping
    |
Clinical audit
    |
Commitment
```

so that an LLM response is never treated as the final patient record without additional deterministic terminology, mapping and audit controls.

Reviewing and Understanding a Patient History

Open Clinical History presents a patient's medical history as a **longitudinal timeline combined with an anatomical visualisation**.

The aim is to make a large clinical history easier to scan while preserving access to:

- the chronological sequence
- the original source evidence
- SNOMED CT coding
- anatomical involvement
- date uncertainty
- clinical status
- provenance

The patient-history viewer is:

```
/patient.php
```

A history is opened by selecting a patient from the patient search screen.

---

1. Select a Patient

The Home page displays the patients currently stored in Open Clinical History.

You can search using either:

- patient record number
- patient name

Leave the search field blank to display all patients.

For example:

```
MRN-1001
Joanne Reacher
```

Selecting the main part of a patient card opens that patient's **Health History Review**.

Understanding the Patient Cards

Each patient card provides a quick summary before the full history is opened.

For example:

```
MRN-1001


Joanne Reacher


b. 1967
Female
12 events
2 active
peak: Major
to 2023
```

Record Number

For example:

```
MRN-1001
```

This is the patient's record number used by Open Clinical History to identify the patient.

Patient Name

The patient's stored display name.

If no name has been recorded, the record number is used instead.

Birth Year

For example:

```
b. 1967
```

This is derived from the patient's recorded date of birth.

Sex

For example:

```
Female
```

This is the patient's recorded sex at birth.

It also determines which anatomical asset set Open Clinical History uses where sex-specific artwork is available.

Events

For example:

```
12 events
```

This is the number of committed clinical events currently stored in the patient's longitudinal history.

An event may represent something such as:

- diagnosis
- injury
- procedure
- investigation
- clinical episode
- significant symptom
- treatment-related event

A source document can produce multiple independent clinical events.

Active

For example:

```
2 active
```

This counts clinical events whose current stored clinical status is:

```
active
```

or:

```
ongoing
```

It provides a quick indication that the history contains conditions that have not been recorded as resolved.

It should not be interpreted as a complete assessment of the patient's current health.

Peak Severity

For example:

```
peak: Major
```

This displays the highest event severity currently recorded anywhere in the patient's history.

The severity scale is:

Level	Meaning
None	No severity classification
Minor	Minor clinical severity
Moderate	Moderate clinical severity
Major	Major clinical severity
Critical	Critical clinical severity

An important distinction is:

> **Severity describes the event at the time it occurred.**

It does not necessarily represent the patient's current condition.

For example, a patient may have had a **Critical** event twenty years ago that is now completely resolved.

`to YYYY`

For example:

```
to 2023
```

This provides a quick indication of how far the dated history currently extends.

It is a summary aid rather than a substitute for reviewing the complete timeline.

Import Summary

The:

Import summary

button opens processing and import information for that patient.

It is useful when investigating:

- imported documents
- extraction results
- terminology resolution
- unmatched events
- learning-queue activity

It is not the normal clinical-history viewing screen.

Delete Patient

Delete patient permanently removes the patient and associated history records.

This is a destructive administrative operation and should not be used as part of normal history review.

---

2. Open the Health History Review

Selecting the patient opens the main:

Health History Review

screen.

The page is divided into three principal areas:

```
Patient + time navigation
           |
           v
+-------------------+---+----------------------+
| Anatomical viewer |   | Clinical timeline    |
|                   |   |                      |
| Body visual       | | | Chronological events |
```

```
|                  | | |                     |
+------------------+---+---------------------+
                     ^

                 Timeline scrubber
```

The anatomical viewer and clinical timeline remain synchronised.

As you move through the patient's history:

- the active clinical event changes
- the anatomical image changes
- the relevant body structures are highlighted
- ongoing anatomical involvement is carried forward
- the year axis updates

---

3. Patient Header

The top of the viewer identifies the patient and summarises the record.

For example:

```
History Review

Record  MRN-1001
DOB     14 Jun 1967
Sex     Female
Events  12
Undated 1
Source  history_import
```

Record

The patient record number.

DOB

The patient's recorded date of birth.

Sex

The recorded sex at birth.

Events

The total number of clinical events shown in this history.

Undated

If present:

```
Undated 1
```

means one or more genuine clinical events were extracted for which the source did not provide enough information to place them reliably on the chronological timeline.

Open Clinical History does **not** invent a date simply so these events can be positioned.

Source

For imported patients you may see:

```
Source history_import
```

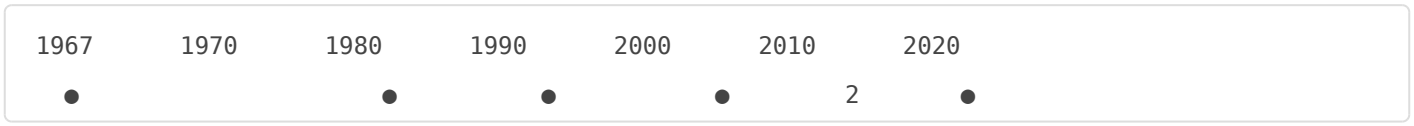
This identifies the system or workflow through which the patient record was created.

---

4. The Year Axis

Across the top of the screen is a compressed view of the patient's entire clinical history.

For example:



Each marker represents a clinical event or a group of events.

This allows a long history spanning decades to be scanned very quickly.

Selecting an Event from the Axis

Click a marker to jump directly to that part of the history.

If several events occurred very close together, they may be combined into one marker containing a number.

For example:

```
2
```

means that two events occupy approximately the same position on the year axis.

Click the marker repeatedly to step through the events in that group.

The events remain separate clinical records.

They are grouped only for display.

Jump to the Beginning or End

The buttons on either end of the year axis jump directly to:

- the earliest event
- the latest event

This is useful for very long histories.

---

5. Date Precision Matters

Clinical records do not always provide exact dates.

A source might say:

```
21 April 1988
```

but it might instead say:

```
1988
```

or:

```
in her twenties
```

or:

```
approximately 10 years ago
```

or provide no date at all.

Open Clinical History deliberately preserves that uncertainty.

It does **not** turn every event into an artificial exact date.

Exact Dates

An event for which the source provides an exact day is represented as a normal filled marker.

For example:

```
21/04/1988
```

means the source actually supplied that date.

Approximate Dates

An event whose exact date is not known is visually different.

Approximate or derived timing uses a **violet uncertainty style** rather than the confident event marker.

A hollow marker means:

> The event belongs approximately here, but the source did not provide an exact day.

Date Ranges

Where the source describes a period rather than a precise date, the year axis can display a horizontal uncertainty span.

Conceptually:

```
1980            1990

      |----------|
        event
```

```
        sometime
        in here
```

The width of the span represents how uncertain the timing is.

The midpoint may be used internally to position the event visually, but:

> **The midpoint is not presented as the event's clinical date.**

Source Wording Is Preserved

Where possible, the viewer displays the source's own description of the timing.

For example:

```
aged 12
```

is more informative than silently converting it to an apparently exact calendar date.

The viewer may additionally calculate the corresponding approximate age or year range.

---

6. Undated Events

Sometimes the source establishes that an event happened but gives no meaningful indication of when.

For example:

```
Previous appendicectomy.
```

with no date, age or relative timing.

Open Clinical History does not position this event at an arbitrary point in the patient's life.

Instead, undated events are placed in a separate:

```
Undated
```

section.

The year axis also provides an **undated** indicator when these events exist.

This distinction is deliberate:

> Not knowing when something happened is not the same as the event not happening.

Why Undated Events Are Not Carried Forward

An undated event may be marked as an ongoing condition, but Open Clinical History cannot safely determine where it belongs relative to other events.

It therefore does not use an undated event to infer:

```
this condition was already active at this later event
```

because doing so would invent a sequence that the source did not provide.

---

7. The Anatomical Viewer

The left side of the screen displays the patient anatomy relevant to the selected event.

The base anatomical image provides context.

Clinical structures are then layered over it.

For example, an event affecting the left shoulder may highlight:

```
Left shoulder joint
```

on the anatomical model.

Amber: This Event

Amber/orange highlighting represents anatomy directly associated with the currently selected clinical event.

The highlighted layer may pulse to draw attention to the relevant anatomical area.

Conceptually:

```
AMBER
=
involved in this event
```

For example:

```
Severe contusions to left shoulder and thorax

            |
            v

        LEFT SHOULDER
        highlighted
```

Blue: Ongoing from Earlier History

A cool blue highlight represents anatomical involvement that originated with an earlier event and has been recorded as ongoing.

Conceptually:

```
BLUE
=
ongoing from an earlier event
```

This lets the reviewer see both:

```
What happened at this point in the timeline?
```

and:

```
What relevant conditions were already ongoing?
```

without confusing the two.

Blue ongoing layers remain steady rather than pulsing.

Viewer Legend

At the bottom of the anatomical viewer you may see:

```
● 1 this event

● 2 ongoing
```

These counts apply to the currently visible anterior or posterior view.

They are the legend for the anatomical colour scheme.

---

8. Anterior and Posterior Views

Use:

```
ANTERIOR
POSTERIOR
```

to switch between the front and back anatomical views.

Some structures exist only on one view.

If relevant anatomy exists on the opposite side, the viewer displays an indicator such as:

```
2 on posterior →
```

Selecting it flips the model to the other view.

This avoids hiding clinically relevant anatomy simply because the current event initially opened on the opposite view.

---

9. Anatomical Hotspots

Relevant structures can have a `+` hotspot positioned over the body image.

Hover over or select the hotspot to see the structure name.

For example:

```
+
Left shoulder joint
```

Selecting a structure opens additional information about that anatomical layer.

The body itself is intentionally kept visually uncluttered.

Structure names appear only when needed rather than permanently covering the anatomical image.

---

10. Moving Through the Timeline

The clinical timeline appears to the right of the anatomical viewer.

Scroll through the page normally to move through the patient's life.

As each event becomes active:

- its card becomes brighter
- the anatomy updates
- the year-axis marker becomes active
- the vertical scrubber moves
- relevant structures are highlighted

This creates a synchronised view of:

```
WHEN
+
WHAT
+
WHERE
```

for each event.

Timeline Scrubber

The thin vertical control between the anatomy and clinical history provides rapid movement through the record.

Drag it up or down to move through a long patient history quickly.

The displayed date changes as you scrub.

Keyboard users can also use:

```
Arrow Up / Arrow Left     previous event

Arrow Down / Arrow Right  next event

Home                      first event

End                       last event
```

when the timeline scrubber has focus.

---

11. Reading an Event Card

A typical event card contains:

```
21/04/1988

Severe contusions to left shoulder and thorax

Severe contusions to the left shoulder and thorax
from a motor vehicle accident.

● Musculoskeletal 1

View details →
```

The card provides the concise clinical-history view.

Event Date

The date appears above the event title.

If timing is uncertain, an uncertainty badge or range accompanies it.

The viewer is designed so that an approximate date should not look like an exact date.

Event Title

The event title is the short clinical description.

For example:

```
Severe contusions to left shoulder and thorax
```

Summary

The event body contains the concise clinical summary extracted from the source record.

For example:

```
Severe contusions to the left shoulder and thorax
from a motor vehicle accident.
```

Severity

The coloured edge of the event card represents the severity assigned to the event.

The scale is:

```
Minor
Moderate
Major
Critical
```

Higher severity events become progressively warmer/redder.

Again:

> Severity represents the event at the time, not necessarily the patient's current condition.

Same-Day Events

Several independent clinical events can occur on the same date.

For example:

```
21 April 1988
2 events
```

Open Clinical History may visually group them under the same date heading.

They are still separate events with their own:

- clinical meaning
- SNOMED coding
- source evidence
- anatomy
- detail record

The grouping is only a presentation aid.

---

12. Body-System Indicators

Instead of displaying a large list of anatomical structures on every event card, Open Clinical History groups anatomy by body system.

For example:

```
● Skeletal 1

● Cardiovascular 2

● Neurological 3
```

Select a body-system indicator to display the individual structures.

This allows complex events to remain easy to scan.

Anatomy Colours in an Event

Within the anatomy list:

Amber means the structure belongs to the selected event.

Blue means the structure is ongoing from an earlier event.

A muted structure may belong to the opposite anterior/posterior view.

---

13. View Details

Select:

View details →

to open the full encounter record.

This is where the concise timeline event can be inspected in greater depth.

A typical detail panel contains:

```
ENCOUNTER DETAIL

Severe contusions to left shoulder and thorax

21/04/1988
Age ~21
Resolved
```

followed by expandable clinical sections.

Encounter Header

The top of the panel can show:

- event title
- event date or date range
- approximate age
- severity
- clinical status

For example:

```
21/04/1988

Age ~21

Resolved
```

The approximate age is calculated using the patient's recorded date of birth.

The `~` indicates that it should not be interpreted as greater precision than the available dates support.

---

14. Summary

The **Summary** section contains the concise description associated with the clinical event.

For example:

```
Severe contusions to the left shoulder and thorax
from a motor vehicle accident.
```

This is the structured summary used by the longitudinal history.

---

15. Source Evidence

Where available, the event retains evidence from the source document.

For example:

```
SOURCE EVIDENCE

Severe contusions to left shoulder and thorax
```

This is important because it lets the reviewer compare:

```
Structured clinical event

        against

Source information that supported it
```

The event should not be considered independent of its provenance.

---

16. Dating

Open the **Dating** section to understand exactly how Open Clinical History decided where the event belongs in time.

For an exact date you may see:

```
DATING  exact day

Recorded as  21/04/1988

Resolves to  21/04/1988

1988-04-21 ──────────── 1988-04-21

The source stated this date.
```

Recorded As

```
Recorded as
```

shows the timing information as represented by the source.

Resolves To

```
Resolves to
```

shows how Open Clinical History represents that information internally.

For an exact day these will usually be identical.

For an uncertain date, the result may instead be a range.

Dating Interval

Every event is treated conceptually as a time interval.

An exact date has:

```
earliest date = latest date
```

For example:

```
1988-04-21 → 1988-04-21
```

An approximate event may instead be:

```
1988-01-01 → 1988-12-31
```

or a wider interval derived from the information available in the source.

This prevents the system from pretending that uncertain clinical timing is more precise than it really is.

Date Derivation

The detail panel also explains **why** the date has that range.

Examples include:

```
The source stated this date.
```

```
The source gave part of a date.

Derived from an age given in the source.

Derived from a life stage described in the source.

Stated relative to the date of the source document.

No dating information was recorded.
```

This makes derived chronology inspectable rather than opaque.

---

17. Anatomy

The **Anatomy** section lists the structures associated with the event.

For example:

```
ANATOMY  1 this event

INVOLVED IN THIS EVENT

● Left shoulder joint
```

Selecting a structure allows it to be inspected on the anatomical viewer.

If relevant structures from earlier ongoing conditions also exist, they appear separately as:

```
Ongoing elsewhere
```

This prevents an old ongoing condition from being mistaken for anatomy caused by the current encounter.

No Structures Recorded

Some valid clinical events do not have a useful anatomical representation.

For example:

- systemic conditions
- administrative clinical events
- some investigations
- some disorders without a useful local anatomical layer

In these cases the viewer may show:

```
No structures recorded
```

This does **not** mean the event itself is invalid.

The anatomical visualisation is supplementary to the clinical record.

---

18. Record

The **Record** section exposes the structured information behind the event.

For example:

```
Encounter ID      EV-3045

Setting           Imported from history document

Coded as          SNOMED CT 125667009
                  Contusion (disorder)

Date precision    Exact day

Date derivation   explicit

Interval          1988-04-21 → 1988-04-21

Outcome           resolved

Provenance        Extracted from source document
                  — not clinician-verified
```

This is particularly useful when reviewing how Open Clinical History interpreted an imported record.

Encounter ID

For example:

```
EV-3045
```

is the Open Clinical History identifier for the individual clinical event.

Setting

This describes the context known for the event.

For imported historical information you may see:

```
Imported from history document
```

Coded As

For example:

```
SNOMED CT 125667009 — Contusion (disorder)
```

This is the structured clinical terminology associated with the event.

Open Clinical History resolves terminology against its local SNOMED CT database.

Date Precision

This explains the precision actually supported by the source.

Examples include:

```
Exact day
Month
Year
Approximate
Not recorded
```

Do not interpret an approximate event as though its displayed timeline position were an exact clinical date.

Date Derivation

This is the machine-readable explanation of how the date was obtained.

For example:

```
explicit
```

means the source directly supplied the date.

Other values indicate that the date or range was derived from less precise information.

Interval

The stored lower and upper temporal bounds.

For an exact event:

```
1988-04-21 → 1988-04-21
```

For an approximate event these values may span months or years.

Outcome

For example:

```
resolved
```

describes the recorded clinical outcome or status of the event.

Other events may remain:

```
active
ongoing
```

depending on the information present in the clinical history.

Severity at Presentation

Where recorded, the detail view may also display the severity of the event.

Remember:

> This describes severity at that clinical episode, not necessarily present-day severity.

---

19. Provenance

One of the most important fields in the viewer is:

```
Provenance
```

For an imported event you may see:

```
Extracted from source document — not clinician-verified
```

This is deliberate.

It tells the reviewer that:

1. the event originated from an imported source document;
2. Open Clinical History extracted and structured the information; and

3. a clinician has not subsequently confirmed that structured representation.

If an event is later verified, the provenance can identify that verification instead.

What "Not Clinician-Verified" Means

It does **not** mean the original source information is false.

It means:

> The transformation from the source material into this structured Open Clinical History event has not been independently confirmed by a clinician.

That distinction matters.

For example:

```
Original medical letter
        |
        v
Open Clinical History extraction
        |
        v
SNOMED classification
        |
        v
Anatomical mapping
        |
        v
Displayed clinical event
```

Until a human clinical verification step occurs, the resulting structured representation should retain that provenance.

---

20. Clinical Components

Some encounters contain several clinically meaningful components.

Where available, the detail panel can display them separately under:

Clinical components

For example, one encounter may contain:

```
Diagnosis
Procedure
Investigation
Complication
```

Each component can retain its own:

- title
- summary
- terminology coding
- anatomical structures

This allows the timeline to remain readable without losing the internal complexity of a clinical encounter.

---

21. Clinical Threads

Where explicit relationships exist between events, the detail panel can show a **Clinical thread**.

Conceptually:

```
Earlier event
      |
      | caused by
      v
Current event
      |
      | led to
```

```
      v
Later event
```

This allows related events to be navigated as a clinical sequence rather than merely by calendar date.

This section appears only where those relationships have been recorded.

---

22. Understanding the Anatomical Visualisation

The anatomy should be interpreted as a **visual index into the clinical history**, not as a diagnostic image.

A highlighted shoulder means:

> This event has been mapped to an installed anatomical layer representing the shoulder.

It does not mean:

- the PNG shows the precise lesion
- the highlighted area represents lesion size
- the brightness represents tissue damage
- the system has interpreted radiology
- the anatomy replaces the underlying clinical evidence

The clinical event and source evidence remain authoritative context.

---

23. Understanding Ongoing Anatomy

When an earlier event has anatomical structures marked as chronic or ongoing, those structures can remain visible while later events are reviewed.

For example:

```
2010
Chronic left knee condition
        |
        v


2015
Unrelated respiratory event
        |
        +--> respiratory anatomy in amber
        |
        +--> left knee remains blue
```

The visualisation therefore provides context about what was already part of the patient's ongoing history at that point.

It should not be interpreted as evidence that the earlier condition was actively symptomatic on every later date.

---

24. A Useful Way to Read a History

For a long patient record, a practical review sequence is:

```
1. Scan the patient summary
        |
        v
2. Look across the year axis
        |
        v
3. Identify major/critical periods
        |
        v
```

```
4. Scroll through the clinical timeline
        |
        v
5. Watch anatomical changes
        |
        v
6. Note ongoing conditions
        |
        v
7. Open important encounters
        |
        v
8. Check source evidence
        |
        v
9. Check dating precision
        |
        v
10. Check SNOMED coding and provenance
```

This provides both a rapid overview and a route into the underlying detail.

---

25. Example Review

Consider the following event:

```
21/04/1988

Severe contusions to left shoulder and thorax

Resolved
```

The viewer may show the left shoulder highlighted.

Opening **View details** reveals:

```
Summary
```

```
Severe contusions to the left shoulder and thorax
from a motor vehicle accident.
```

The reviewer can then establish:

When did it happen?

```
21/04/1988
```

The Dating section confirms:

```
Exact day
```

and:

```
The source stated this date.
```

How old was the patient?

```
Age ~21
```

calculated from the recorded DOB.

What anatomy was involved?

```
Left shoulder joint
```

What was it coded as?

```
SNOMED CT 125667009 — Contusion (disorder)
```

What happened afterwards?

```
Outcome: resolved
```

Where did the information come from?

```
Extracted from source document — not clinician-verified
```

This is the intended review model:

```
EVENT
+
TIME
+
ANATOMY
+
CODING
+
SOURCE
+
PROVENANCE
```

rather than simply presenting an AI-generated summary.

---

26. What the Viewer Deliberately Does Not Do

The patient-history viewer deliberately avoids several potentially misleading behaviours.

It does not invent exact dates

If the source says only:

```
1988
```

the interface does not silently turn that into:

```
01/01/1988
```

and present it as fact.

It does not hide undated events

An event without a known date remains visible.

It is simply kept outside the dated chronology.

It does not pretend ongoing anatomy belongs to the current event

Current-event anatomy and carried-forward anatomy use different visual treatments.

It does not hide provenance

Imported structured information can explicitly state that it has not been clinician-verified.

It does not require an anatomical image for every event

A clinically valid event can exist without a body-layer mapping.

It does not treat the body image as the clinical record

The anatomy is a navigation and comprehension aid.

The clinical event, evidence, terminology and provenance remain accessible separately.

27. Important Interpretation Notes

When reviewing an Open Clinical History timeline:

A timeline position is not always an exact date

Check the Dating section when precision matters.

A highlighted body structure is a mapping

It represents the anatomy associated with the clinical concept, not a diagnostic image.

Blue does not mean a new event

Blue indicates relevant anatomy carried forward from an earlier ongoing condition.

Severity is historical

A Critical event may be decades old and resolved.

"Active" is based on the structured record

It reflects what has been recorded as active or ongoing, not a new clinical assessment.

An AI-extracted event should retain its provenance

If the detail says:

```
not clinician-verified
```

that should be considered when relying on the structured representation.

Source evidence matters

Where something appears surprising, open **View details** and compare the event against its source evidence and dating information.

Summary

The Open Clinical History viewer is designed around five questions:

```
WHEN did it happen?

WHAT happened?

WHERE in the body was involved?

WHAT remained ongoing?

HOW do we know?
```

These are represented through:

```
WHEN
Timeline and dating intervals

WHAT
Clinical event cards

WHERE
Anatomical visualisation

ONGOING
```

```
Carried-forward anatomical context


HOW DO WE KNOW?
Source evidence, SNOMED coding and provenance
```

The aim is not merely to make a health record visually attractive.

It is to make a complex longitudinal history **quick to scan while preserving the uncertainty, evidence and provenance needed to understand what the visualisation actually means**.