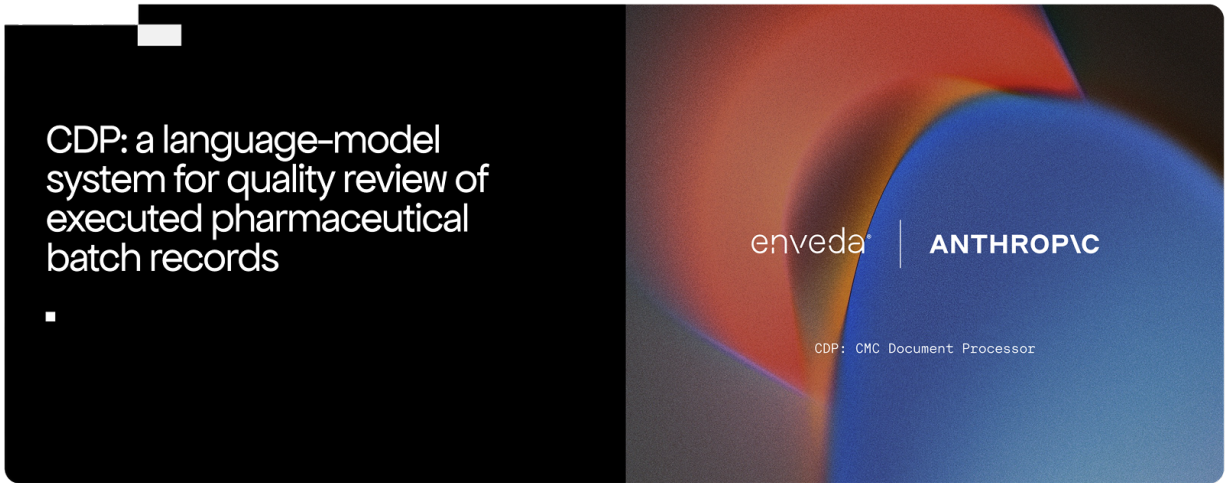




Supplementary Information

CDP: a language-model system for quality review of executed pharmaceutical batch records

Supplementary Notes 1–12, Supplementary Tables 1–4, Supplementary Figs 1–6, Appendices A–C and Appendix Tables 1–2

Contents

- **Supplementary Note 1** | Purpose, scope and intended use
- **Supplementary Note 2** | System description and boundaries
- **Supplementary Note 3** | Installed components
- **Supplementary Note 4** | Infrastructure and environments
- **Supplementary Note 5** | Configuration items and configuration management
- **Supplementary Note 6** | Interfaces
- **Supplementary Note 7** | Functional description: from upload to advisory outcome
- **Supplementary Note 8** | Data integrity and the audit trail
- **Supplementary Note 9** | Security and access
- **Supplementary Note 10** | Operational controls
- **Supplementary Note 11** | Known limitations
- **Supplementary Note 12** | Mapping to regulatory expectations
- **Appendices**

Supplementary Note 1 | Purpose, scope and intended use

1.1 Purpose

These notes describe CDP as it was installed, configured and operated for the study reported in the main text. They follow the structure of an installation qualification. They cover what the system is and where its boundaries lie, what it is made of, where it runs, how it is configured, how it connects to other systems, what it does, how its records are kept and how it is controlled. Each section states the installed state and the record that shows it. The study design and statistics are in the main text and are not repeated here.

1.2 Intended use

CDP is an advisory aid for the quality review of GMP documents supplied by external manufacturers and laboratories, chiefly executed batch records. It reads each document, checks it against the documents that govern it, such as the master batch record and the specification, and posts comments and an advisory outcome beside the file. People review and disposition every comment, and the quality unit decides every batch. CDP does not approve, reject, release or close any record.

1.3 GxP role and status

CDP has not been validated as a GMP computerized system, and these Notes are not a validation package. An internal assessment classes CDP as a low-risk advisory tool. It is designed and operated as a non-critical application in the sense of draft EU GMP Annex 22¹⁵. Supplementary Note 12 maps its controls to GMP expectations and states which are enforced in code, which by the hosting platform and which by written procedure.

1.4 Scope

In scope:

- CDP orchestration application and workers
- CDP configuration, held as versioned files in the ECM
- CDP interfaces to the system of record, the AI model provider and the secrets vault
- CDP written audit logs

Out of scope:

- The ECM platform that serves as the system of record.
- The AI model provider's service
- The two human reviews that sit outside CDP: the manufacturer's quality review before a record reaches CDP, and the quality unit's disposition after CDP review.

Some components were built but were not part of the evaluated configuration, such as routing and renaming of files and email intake. Supplementary Note 11 lists them.

1.5 Configuration described

These notes describe the configuration in production for the CDP pass reported in the main text. Supplementary Table 1 gives its main parameters.

Supplementary Table 1 | Evaluated configuration

Parameter	Value
AI model and refusal fallback	Claude Opus 4.7, with Claude Sonnet 4.6 configured as fallback (not invoked in the pass)
Transport	Inline PDF document blocks
Unit of the evaluated document	Windows of at most 40 pages and 8 MB, packed by measured page bytes
References	MBR and specification sent with every window (Supplementary Note 7)
Prompt cache	5-minute lifetime
Output budget per window	32,000 tokens
Reasoning	Adaptive thinking, effort "high"
Prompts	The production prompts on the day of the run, identified in the ECM's version history
Event poll interval	60 s

1.6 Sources for the installed state

- The configuration register. This is CDP's installation-qualification register (Supplementary Note 5.5).
- The live configuration files in the ECM, read for each tenant.
- A snapshot of CDP's code taken after the study
- Production records: the AI-call log, the review log and the comment-audit ledger.

1.7 Relationship to the published code

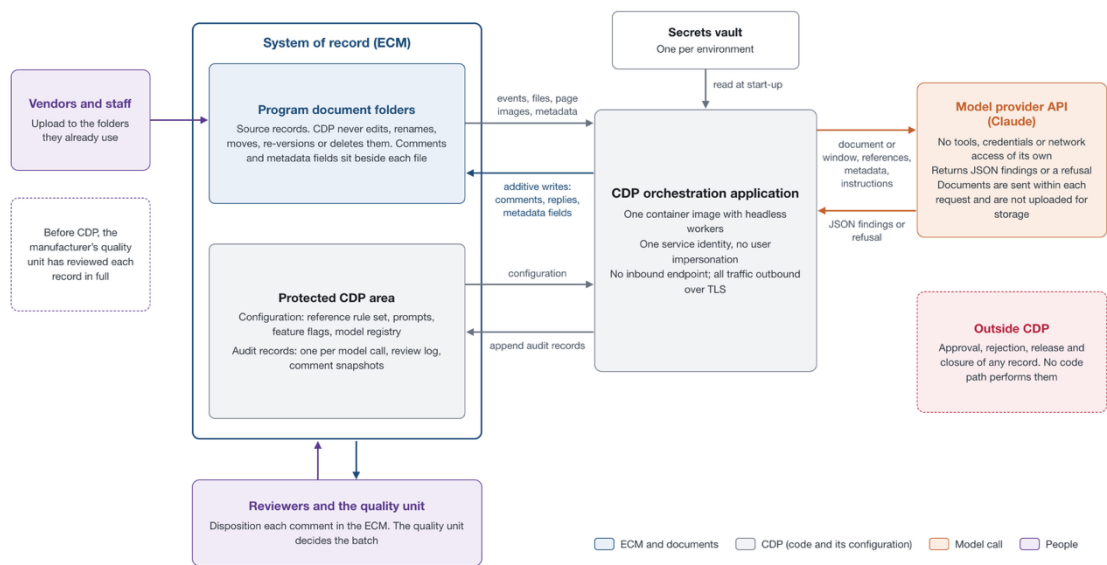
CDP is an internal system and its code is not published. The code released with this paper, open-doc-review (Code availability), is a separate open-source implementation of CDP's core design, written to work with any ECM. It is smaller than CDP and is not the code that produced the study results. Its documentation covers installation, configuration, writing review prompts and a template for local validation.

Capability	CDP (evaluated)	open-doc-review
Classification and extraction into metadata	Yes	Yes
Finding review reference documents from extracted identifiers	Yes, with a reference rule set, a resolution ladder and confidence caps (Supplementary Note 7)	Yes, by matching the fields declared for each document type
Page-windowed review of long PDFs	Yes	Yes

Capability	CDP (evaluated)	open-doc-review
Findings written beside the document; document content never edited, moved, renamed or deleted	Yes	Yes
Audit record for every AI model call	Yes, stored in the ECM	Yes, as a local file
Refusal fallback	Set per use case in the AI model registry	The provider's fallback, which can be switched off
Run without writing review results to the ECM	Capture-only mode	Dry-run mode
Evaluation against documents with known findings	Regression corpus (Supplementary Note 10.4)	A built-in evaluation command
When a document changes	Delta re-check of the revised version against the open comments, on request	Full re-review of each new version, and of a document whose missing reference arrives
Existing comments	People's comments are kept on re-review; CDP's own earlier comments are replaced	Comments are never deleted
Coverage ledger, cross-window reconciliation and identifier registries	Yes	No
ECM connections	One cloud ECM	One cloud ECM and a local folder supported; one further ECM experimental; example adapters for others, and an interface for writing new ones

Supplementary Note 2 | System description and boundaries

CDP is an orchestration application that sits between the ECM and an AI model API (Supplementary Fig. 1). The AI model has no tools, no credentials and no network access of its own. It receives a request and returns structured JSON. Everything that touches a GMP record is done by the orchestration application, under one service identity. The application writes only comments, replies, metadata fields, audit records and per-program analytics files, and the source records are never modified.



Supplementary Fig. 1 | System boundaries and data flows. The CDP orchestration application (grey) is the only component connected to both the system of record (blue) and the AI model provider (orange). It reads documents, events, metadata and configuration from the ECM and writes back only additive content: comments, replies, metadata fields and audit records. The AI model receives a document or window with its references and instructions, and returns JSON findings or a refusal. Approval, rejection, release and closure of records sit outside CDP.

2.1 What crosses each boundary

Connected system	CDP reads or sends	CDP writes or receives	Limit
System of record (ECM)	File versions, the original PDF and page images, metadata, comments, the event stream and CDP's configuration files	Metadata-template instances, comments and replies, and append-style audit files in a protected area	No path edits document content. Renaming, moving and re-versioning are switched off
AI model provider API	The document or one window of it, its review reference documents, extracted metadata and the instructions	JSON findings or a refusal	In the evaluated configuration, documents are sent within each request and are not uploaded for storage. The provider's standard data-retention terms apply, with no zero-data-retention agreement

Connected system	CDP reads or sends	CDP writes or receives	Limit
Secrets vault	Secrets, at start-up, through the workload's managed identity	—	One vault per environment

Copies of documents leave the system of record at every stage that reads them. These stages are classification, extraction, the row-level registries, the lot index, genealogy and review.

2.2 People around the system

Vendors and staff upload documents to ECM per routine use. Reviews can also be queued from the team's work-tracking tool, which receives a short summary of the findings. CDP never changes the status of an item there. Before a record reaches CDP, the manufacturer's quality unit has reviewed it in full. After CDP has run, reviewers disposition each comment in the ECM. The sponsor's quality unit then carries out a final review of the record and decides the batch's disposition.

2.3 Design principles and how they are enforced

#	Principle	How the installed system enforces it
1	The source record is not changed	The application writes only metadata, comments, replies and append-style audit files. Renaming, moving and re-versioning of files are disabled in every environment
2	The AI model reads and comments, and people decide	No code path accepts, rejects, releases or closes a record. The outcome is advisory, and a named person records acceptance outside CDP
3	Every page is either reviewed or reported as not reviewed	Each page is sent to the AI model in exactly one window or is declared unread with a reason. Chart pages the AI model reports it did not evaluate are declared in a notice on the document
4	Code enforces the limits on what the AI model returns	Confidence caps, downgrade-only reconciliation, window sources stamped by code and conversion to absolute page numbers are all done in code
5	Each record is checked against the right review reference documents	Linking identifiers are checked against a registry built only from the documents that define them. The review header shows how each reference was found
6	Configuration is held as versioned files in the system of record	Prompts, the reference rule set, feature flags and the AI model registry are files in a protected configuration area (Supplementary Note 5)
7	Failures are visible on the document	A dependency outage freezes the event cursor. A failed file is held with a comment naming the step, and any page not read is named on the document
8	AI model calls can be attributed and reconstructed	Every AI model call in review, extraction and the registries is logged with the AI model, the instructions, the response and token use (Supplementary Note 8)
9	Run-to-run variation is measured	Sampling parameters cannot be set on the AI models used and no seed is available, so findings vary between identical runs ²⁵ . The evaluation measures this variation (main text, Configuration management and pre-release testing)
10	Environments are separate	Each environment has its own tenant, vault, configuration copy and deployment. A start-up check refuses to run against the wrong tenant (Supplementary Note 4)

Supplementary Note 3 | Installed components

The installed system has ten components (Supplementary Table 2). Components that were built but are switched off in the evaluated configuration, such as email intake, are listed in Supplementary Note 11.

Supplementary Table 2 | Installed components

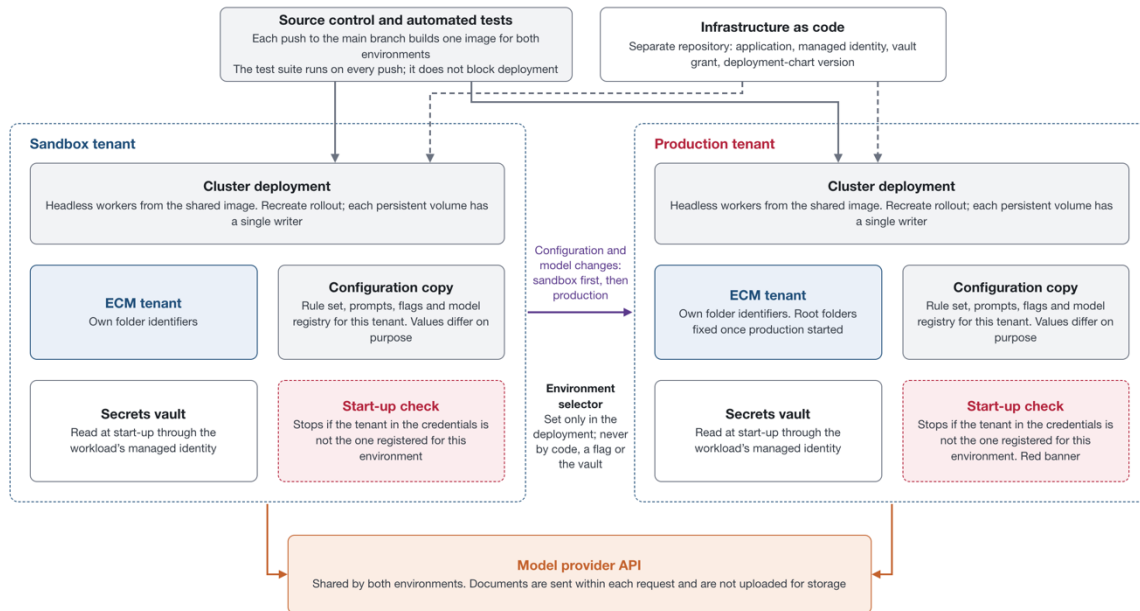
Component	What it is	Inputs and outputs	Controls
C1 System of record	Validated cloud ECM platform. Its own version history, metadata history and comment attribution are unchanged by CDP	Gives file versions, the original PDF and page images, metadata templates, comments, the event stream and configuration files. Accepts metadata, comments and replies, and audit files	Never accepts edits to document content. Rename, move and version paths are off
C2 Event intake	Single event reader	Upload, new-version and metadata-update events in. Ordered work items for each file out	Durable cursor, de-duplication by event ID, cursor frozen while a dependency is down. No event lost or duplicated across a restart
C3 Orchestrator	One container image with headless workers	Work items, configuration and secrets in. AI model requests, allow-listed ECM writes and audit records out	The only component connected to both C1 and C4. One service identity, with no user impersonation
C4 AI model API	The provider's Messages API, with prompt caching	System prompt, references, metadata block and one document window in. JSON findings, coverage and charts, or a refusal, out	No tools, no credentials and no write path
C5 Configuration store	Protected configuration area inside C1	Reference rule set, prompts, feature flags, AI model registry, classification prompt, filename patterns and master-data registries	Versioned files. Re-read after 2, 5 or 15 minutes (Supplementary Note 5.1)
C6 Audit store	Append-style files in C1, buffered locally	One record per AI model call, one review-log row per run, comment snapshots, the comment-action archive, ingest failures and poller health	Dead-letter queue with replay (Supplementary Note 8)
C7 Secrets vault	One per environment	Secrets out, at start-up	Read only through the workload's managed identity
C8 Container platform	Managed container orchestration	—	Recreate rollout. Persistent volume with a single writer
C9 Build and test pipeline	Automated test suite run in the container-image build	—	Does not block a deployment
C10 Observability	Per-cycle call census, health files and dashboards	—	A watchdog checks the health files

3.1 Scale of use

Item	Value
Automated tests	About 3,000, run on every push (Supplementary Note 10)

Item	Value
AI model use cases	21 in the AI model registry, each with a primary AI model and a refusal fallback. A few calls do not read their registry entry and have no fallback (Supplementary Note 6.2)
Document types	27 in the classifier's controlled list. Reference rules for about 23 types, with 52 reference edges
Production review volume	17,659 comments from 1,563 reviews of 462 files in the first six weeks of production. The median review had 10 comments, and batch-record reviews averaged 22.0
Corpus classified	13,362 files in the initial migration, 98.9% tagged. About 19,000 documents in all

Supplementary Note 4 | Infrastructure and environments



Supplementary Fig. 2 | Environments. Sandbox and production are separate tenants of the ECM, each with its own cluster deployment, configuration copy and secrets vault. Both run the same container image, built from the main branch, and the infrastructure is declared in a separate repository. The environment is chosen only in the deployment, and a start-up check stops the process if its credentials belong to another tenant. Configuration and AI model changes are applied in the sandbox first. Both environments use the same AI model provider.

4.1 Hosting

- A single container image with one entry point serves each worker role. A setting in the deployment selects the role.
- The workers are headless. They have no inbound endpoint, and liveness is shown by health files and a watchdog. All traffic is outbound over TLS on port 443, to the ECM, the AI model API and the vault.
- The event worker keeps its runtime state on a mounted persistent volume. This state is the event cursor, local log buffers and dead letters. Each volume has a single writer, and a recreate rollout makes the old instance release the volume before the new one starts.
- Memory is bounded when documents are reviewed concurrently.

4.2 Infrastructure as code

A separate repository declares the application, its managed identity, that identity's read grant on the environment's vault and the deployment-chart version. A production deployment needs three things:

the production stack must exist, the production vault must be populated, and the production tenant identifier must be registered in code. Without the registered identifier, the process refuses to start.

4.3 Environments

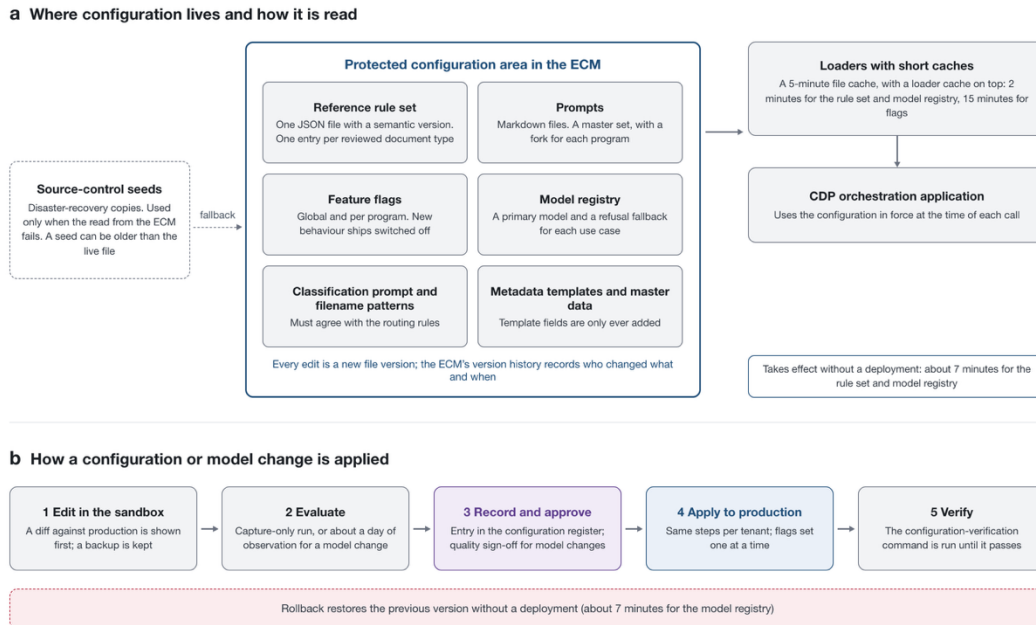
Sandbox and production are separate tenants of the ECM (Supplementary Fig. 2). Each has its own vault, folder identifiers, configuration copy and cluster deployment.

A single environment selector chooses the environment. It is set only in the deployment, and code, a command-line flag or the vault cannot set or override it. At start-up the process prints a banner, red for production, and loads its secrets from the matching vault. It stops if the tenant identifier in its credentials does not match the one registered for that environment.

4.4 Deployment of code

Each push to the main branch of source control builds one container image and deploys it to both environments. The full automated test suite runs on every push in a separate pipeline and does not block a deployment. Inside the suite, a check confirms that the classification prompt, the routing rules and the filename patterns agree with one another. A read-only report shows, for each tenant, the code version, configuration and flags in force. Configuration is deployed separately from code (Supplementary Note 5).

Supplementary Note 5 | Configuration items and configuration management



Supplementary Fig. 3 | Configuration items and how they change. a, CDP's runtime configuration is held as versioned files in a protected area of the ECM and read through short-lived caches. Source-control copies are used only if the read from the ECM fails. b, A change is made in the sandbox first, evaluated, recorded in the configuration register and then applied to production. AI model changes need quality sign-off. Rollback restores the previous version without a deployment.

5.1 Where configuration is held

CDP reads most of its runtime configuration from the system of record (Supplementary Fig. 3a). Each change is a versioned file edit there, and the platform's version history and metadata audit trail record who made it and when.

Files are read through two caches: a 5-minute file cache, with a loader cache on top of it of 2 minutes for the reference rule set and the AI model registry, and 15 minutes for feature flags. An edit therefore takes effect without a deployment: within about 7 minutes for the rule set and the AI model registry, and within about 20 minutes for flags.

The copies in source control are seeds for disaster recovery. If the read from the system of record fails, the loader uses the seed, not the last copy it read successfully. A change merged into source control changes nothing in either environment until it is pushed to that environment.

5.2 Configuration items

Item	Form (evaluated configuration)	How it is changed
Reference rule set	One versioned JSON file. Rules for about 23 document types, with 52 reference edges (Supplementary Note 7)	Upload tool that first shows a diff against production. Targeted edits for single keys
Review prompts	Markdown files. A master set, with a fork for each program made at onboarding. The version in force on any day is found from the ECM's version history (Supplementary Note 8.1)	Prompt sync, run first as a dry run and then applied
Feature flags	95 settings, global and per program. Values differ between tenants on purpose	Version-checked edit of a single key, verified by re-reading. A tenant-diff command shows the differences
AI model registry	A primary AI model and a refusal fallback for each of 21 use cases, with the rationale for each choice	Edit tool for each use case, with a backup and a revert
Classification prompt and filename patterns	Must agree with the routing rules. A test checks the source-control copies, and a configuration-verification command checks the live copies	Targeted edit, or upload of the whole file after a diff
Metadata templates	Fields are only ever added	Idempotent set-up with a dry run, an apply step and a check of the field count

5.3 How a change is applied

A change is applied to each tenant in turn, sandbox first and then production (Supplementary Fig. 3b). The steps are:

1. Push the changed configuration.
2. Update the metadata templates.
3. Sync the prompts.
4. Set flags deliberately, one at a time.
5. Run any backfills. They are journaled and fill only empty fields.
6. Run the configuration-verification command until it passes.

An AI model change follows a set procedure. The sandbox registry is edited first, with a backup. The change then runs for about a day while the team watches the resolved AI model, the stop reasons, the fallback rate and the backlog. It is entered in the configuration register, which carries a criterion for quality sign-off, and only then applied to production. A rollback restores the previous values and is live within about 7 minutes, with no deployment.

AI model changes, and flags that can shift review output, are evaluated in capture-only runs before they are switched on (Supplementary Note 10).

5.4 Feature flags

New behaviour ships behind a switch that is off. It is turned on per tenant after a check in the sandbox. Once a switch is on in every tenant, its default in code and in the seed is updated to match, and a test checks that the two agree. In production, every enabled setting must have an effect that can be shown, and switches that depend on each other are set together. Every flag has an entry in a test-evidence catalogue that names its default and the tests that cover it.

In a production snapshot taken shortly before the evaluated pass, the following were on:

- whole-document reconciliation
- window coverage reporting
- page anchors
- candidate matching
- the field-confidence pass
- comment preservation

Review on upload, routing and renaming were off.

5.5 The configuration register

The configuration register is CDP's installation-qualification register. It lists every configured resource in the order in which it is set up:

1. the day-0 bootstrap
2. environment and authentication
3. entry points
4. master data
5. metadata templates
6. folder structures
7. event polling
8. email
9. the points where configuration is synchronized

Each feature has its own block of numbered criteria, each with an evidence type and a result. The register ends with the extraction field definitions and a change history. Once every section has been verified and signed, the completed register is the evidence that the system is installed and configured to its design.

The register is versioned, with a change history, and each significant change adds criteria through deployment instructions. A working rule, which is also a coding rule, requires that any manual verification step found during development or maintenance be added to the register before the change is complete.

Supplementary Note 6 | Interfaces

CDP has three interfaces: the ECM, the AI model provider API, and the secrets vault (Supplementary Fig. 1). This note describes what passes across each one and how failures are handled.

6.1 System of record (ECM)

Events

CDP reads the ECM's enterprise event stream by polling. In the evaluated configuration the poll interval was 60 s. it is set in configuration and limited to 30–3,600 s. Events are grouped by file, so an upload and a later edit of the same file keep their order, and distinct files are processed in parallel, up to eight at a time. The table shows how each event is handled.

Event	What CDP does
Upload	Ingestion: classification, extraction and the registries (Supplementary Note 7.1). No review runs, because review on upload is switched off
New version of a file	No automatic action. A person can request a re-review
Edit of a master-data file	The matching set-up step runs again. It is safe to repeat
Re-review requested (metadata field)	Full re-review with comment preservation, or a delta re-check (Supplementary Note 7.6)
Send-back requested (metadata field)	A draft email to the vendor is posted as a comment (Supplementary Note 10.1)
Reclassify requested (metadata field)	Reclassification and re-extraction. Reviews wait until it finishes
Identifier confirmed (metadata field)	The confirmed identifier is admitted to the identifier registry

Dependency checks

Before each cycle CDP checks that the ECM and the AI model API are reachable. If either is down, the cycle is skipped and the event cursor does not move, so no event is lost. Files that failed while a dependency was down are kept and run first in the next healthy cycle. Other failures are held with a comment on the file and a failure record (Supplementary Note 7.1). A kill switch pauses intake, which resumes where it stopped. The cursor and the retry list are saved on the worker's persistent volume.

Reading documents

CDP downloads each PDF once and splits it into windows locally, so a long review never depends on many separate page fetches. Office documents are read only through the ECM's PDF rendition. If the rendition is not ready after a second attempt, CDP posts a comment asking the uploader for a PDF and excludes the file from review.

Writing

CDP writes metadata-template instances, comments and replies, and append-style audit files in its protected area. Each comment post is tried up to three times, and the review log records any post that still failed.

Platform limits and how CDP handles them

Limit	How CDP handles it
AI model API requests are limited to 32 MB after base64 encoding, and PDF input to 600 pages per request	Separate size checks on raw and encoded bytes. Larger documents are sent as PDF page slices at full resolution; only a single page that cannot fit is sent as an image
Requests with many images are limited to 2,000 px per side	Pages are normally sent as native PDF at full resolution. A page image is a fallback, used only for a single page too large to send as PDF, and is rendered at 1,024 × 1,024 to stay within the limit
The ECM has no page-range rendition	The PDF is downloaded once and sliced locally
The ECM's text rendition is empty for scanned PDFs	The body length is checked, with a fallback to images
Deleting a comment in the ECM deletes its replies, the API does not show a reply's parent, and comments have no resolved state	Comment-preservation rules (Supplementary Note 7.6)
A file keeps only its newest 1,000 versions	Append-only logs rotate daily

6.2 AI Model provider API

The AI model has no tools, credentials or network access of its own. Each request carries the document or one window of it, its review reference documents, a block of extracted metadata and the instructions. The AI model returns JSON or a refusal. Appendix Table 2 lists every AI model call, with its trigger, inputs, output contract and failure behavior.

What CDP needs	API feature used
Reading scanned PDFs, including handwriting, tables and charts	PDF document blocks, sent inline in the evaluated pass, with page images as a last resort
One shared prefix of instructions and references across windows	Prompt caching
Reasoning depth	Adaptive thinking at effort "high"
Detecting refusals and cut-off answers	The refusal and output-limit stop reasons
Cost accounting	The input, output and cache token counts returned with each call

Transport

In the evaluated pass, documents were sent inline in each request.

AI model registry and refusal fallback

The AI model registry names a primary AI model and a refusal fallback for each of 21 use cases. Window review falls back only on a refusal; rate limits and overloads are retried on the same AI model. Other calls

also fall back on a rate-limit or server error. A few calls do not read their registry entry and have no fallback, among them the advisory-outcome and trend-narrative calls, which use the window-review AI model. The AI model that answered is recorded for every window. The evaluated pass had no refusals, so the fallback was not used.

Retries

Each window is tried up to four times, with jittered exponential back-off, for transient errors only: rate limits, server errors, overload, timeouts and connection errors. Permanent errors, such as a request that is too large or an answer cut off at the output limit, are reported without a retry. A request that is too large is never shown to a reviewer as "service busy".

6.3 Secrets vault

Secrets are read once, at start-up, through the workload's managed identity, from the vault for that environment. A changed secret takes effect after a restart.

Supplementary Note 7 | Functional description: from upload to advisory outcome

This note describes what CDP does with a document, in the order it happens. Fig. 2 of the main text shows intake and linking, and Fig. 3 shows the review. Intake runs on every upload; review runs only when a person requests it.

7.1 Intake: classification, extraction and registries

Classification

One AI model call assigns the document type and a material category. The 27 types in the controlled list are certificate of analysis, certificate of conformance, executed batch record, analytical raw data, specification, analytical method, stability data, deviation report, change control, shipping record, regulatory submission, quality agreement, development report, validation, manufacturing document, master batch record, safety document, technology transfer, vendor communications, vendor general, memo, vendor SOP, customs document, contract or statement of work, master process instruction, meeting minutes and campaign report. Subtypes, such as a stability protocol or a certificate of analysis for lot release, select the review prompt. The material category is one of drug substance, drug product, intermediate, raw material, reference standard and authentic substance. Vendor titles for an executed record, such as "batch production record", map to one type. Large documents are classified from their opening pages. A type set by a person is never overwritten. Classification runs only as CDP's own logged AI model call.

Extraction

An extraction template for each type lists the fields to read. Every template carries universal fields (lot numbers, vendor document identifier, document date, campaign identifier, program code, vendor product code, compound code and vendor address) and fields specific to the type. The campaign identifier is filled from a registry, never by the AI model. Each field has its own short instruction: the labels to look for, including their equivalents in a second language, where on the page to look, the format and when to leave the field blank. Documents of up to 400 pages are read in windows, and list fields gathered across windows are combined. The PDF is the ground truth: extracted metadata is passed to the review as supplementary information, and the review is told to trust the PDF when the two disagree. After three failed attempts the file is held with a comment.

Registries

Intake fills the registries below. Reviewers can confirm identifiers and edit genealogy.

Registry	Holds	Safeguard
Master data	Programs, vendors, vendor-program assignments and campaigns	An unrecognized vendor is flagged for a person to confirm and is never added automatically

Registry	Holds	Safeguard
Identifier registry	Canonical identifiers, each built only from the document type that defines it (a specification defines its number, a method its method number)	An extracted identifier must match the registry, so an identifier the AI model invented never becomes a link. A new identifier is admitted automatically only at high confidence; otherwise a person confirms it in the file's metadata
Test and stability results	One row per result or timepoint, with its limits	A confidence for each row; reviewers triage rows below high confidence
Batch parameters	Process parameters, in-process results, equipment and raw-material lots from executed records	A confidence for each row
Lot index and genealogy	The lots in each document, and which lots fed a manufactured lot	Reviewers can edit genealogy. A failure never blocks processing

Held files

A file is held when classification or extraction fails, when its program code is missing or does not match its folder, or when its product type does not match. A comment on the file names the failed step and the fix, and a failure record is written to the audit store. The comment is posted even if the failure record cannot be written.

7.2 Finding the review reference documents

A record is reviewed against the documents that govern it, such as the master batch record (MBR), the specification, the methods it cites and the deviations it references. There is no separate graph database. Links are found at review time by metadata queries in the ECM, defined in the reference rule set (Supplementary Note 5.2), which has one entry for each reviewed document type.

What an entry declares

For each reference: its role name as shown to the AI model, the document type it points to, whether it is required, and whether it is strict (left unresolved rather than matched loosely). For each lookup: the query, filled from the evaluated document's extracted fields; a preferred tie-break field; an ordered list of looser fallback queries; whether to look up each item in a list field separately (for example, each cited deviation); the dimensions a candidate must match, such as dosage form; and the size and order of the candidate pool.

Resolution ladder

Each reference is matched as strictly as possible, and the review header shows how each was found. The table lists the matches from strictest to loosest; the pharmacopoeial backstop runs only after every other step has failed.

Match	How	Label shown to the reviewer	Effect on confidence
Exact	Document number and version	[exact]	None
Normalized	Differences in punctuation, spacing or prefix bridged	[normalized] or [separator-bridged]	None

Match	How	Label shown to the reviewer	Effect on confidence
Same document, any version	Highest version of the same document	[exact]	At most Medium where the rule prefers a field, such as the version, that the pick does not match
Secondary identifier	A vendor cover identifier matched to the internal document	[exact]	None
AI model-assisted pick	When several candidates remain, the AI model reads the record and each candidate and picks one or abstains	[ai-picked]	At most Medium
Pharmacopoeial backstop	For cited methods only: a cited method is mapped to a pharmacopoeial chapter, or not	[compendial (in-house)] or [compendial (AI knowledge)]	At most Low
Product fallback	Newest document of the type for the same program and material	[fall-back]	At most Low
Several equal candidates	One is used	[ambiguous]	None
Not found	Listed as missing	—	At most Medium if required

Both AI model-assisted steps may abstain. The AI model pick's confidence never exceeds medium, and on any failure the deterministic pick stands. Lookups are bounded at 300 ECM calls per review. Every reference, and how it was found, is listed in the review's header comment. A reviewer can add up to three references, or exclude one with a reason, in the file's metadata; these are applied after resolution.

Executed batch records

The MBR is found by the MBR number printed on the record, with a tie-break on the vendor document identifier, so that sibling batches sharing one template number each get their own MBR. The specification is found by the cited specification number. Deviation reports are looked up one by one for each cited deviation. These references are optional, because an executed record normally carries its own limits: a missing MBR or specification is not a finding and does not lower confidence. When the MBR found does not match the record's vendor identifier, confidence is capped at Medium. Upstream lot records are not linked (Supplementary Note 11).

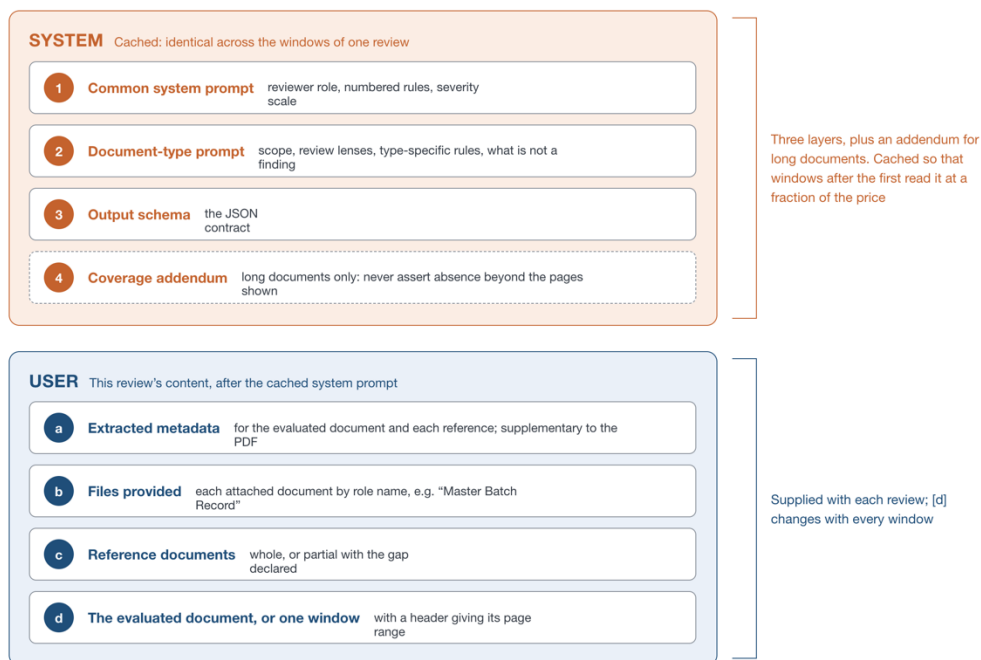
Campaign reports

A campaign report is checked against every executed record it covers. Each record is attached as a short digest of its extracted yields, lots and reconciliation values, and the report's claims are checked against these sources one at a time.

7.3 Review instructions and output

Each review call is built from layers held as files in the ECM (Supplementary Fig. 4): a system prompt with the reviewer role, numbered rules and the severity scale; a prompt for the document type; the output schema; and, for long documents, a coverage addendum. Each file is looked up first in the program's

own copy, then in the master copy. Appendix B gives a redacted template of the system and batch-record prompts.



Supplementary Fig. 4 | Composition of one review call. The cached system part (orange) holds the common system prompt, the prompt for the document type and the output schema, plus the coverage addendum when the document is reviewed in windows. The user part (blue) is supplied with each review: extracted metadata, the list of attached files, the reference documents and the evaluated document or one window of it, which changes with every window.

Rule families

The batch-record rules in force for the evaluated pass are summarized below in paraphrase.

Rule family	Rule (paraphrased)
Advisory scope	Advisory only; a person makes every decision
Page evidence	Cite the page and the value exactly as printed; never guess a page
PDF as ground truth	When extracted metadata and the PDF disagree, trust the PDF
Missing references	A missing reference is a finding only when the document itself cites something absent or contradictory
Signatures	CDP reviews executed records before the vendor's quality sign-off, when signatures, initials and their dates are often not yet complete. These are therefore left to the human reviewer and are not raised by CDP. The integrity of recorded data and the two-person check on critical values stay in scope
Limits	A value inside its limit is not a finding; a value that crosses its limit is a finding at full severity
Totals	Where components and a total are recorded, add them up; a total that differs is a finding
Corrections	A documented correction with a reason is complete; judge whether the value that now stands is legible
For-information results	Not judged against a limit borrowed from another document

Rule family	Rule (paraphrased)
Chronology	Raise a chronology finding only when two events are causally linked
Recurring issues	One consolidated finding for a recurring class of issue
Handwritten changes to printed targets	Always reported, as informational
Branches marked not applicable	Blank entries inside a properly marked branch are not findings
Derived values	Where inputs are recorded and the review reference document gives a rate, duration or ratio, compute and compare
Bilingual records	Values that disagree between language versions are findings; the bilingual layout is not
Uncertainty	A potential excursion that cannot be confirmed or ruled out is raised as "unconfirmed: reviewer to verify"
Window scope	Never assert that something is absent beyond the pages shown

Output

The AI model returns one JSON object per call (Appendix Table 1): findings with a title, body, grade, page and the references used; mitigations, such as a recorded deviation, with their identifier, page and status; coverage entries naming what the pages document; and pointers to chart pages it did not read against a limit. The review call never returns the advisory outcome or a confidence. CDP converts page numbers to absolute pages and posts each finding as a comment of the following form.

```
[MAJOR • T] Charge time implies rate above MBR maximum
Step 7 records a solvent charge whose duration implies a rate about 20% above
the MBR maximum (PDF p.12, printed p.88); unconfirmed: reviewer to verify.
l, Reconciled from BLOCKER: <reason, with page> (only if lowered)
p.12
.v3
```

Here T is the review category and v3 is the file version the comment was raised against. Each review also posts a header comment listing the references and how each was found, and a recommendation comment with the advisory outcome, its confidence, a one-sentence rationale and the risk tier. System notices, marked [System], report pages not read, chart pages not evaluated and references found loosely or not found.

7.4 Long documents

The executed batch records in this study run to hundreds of scanned, mostly handwritten, bilingual pages. CDP reads each document in windows small enough to send every page as native PDF at full resolution, sends the review reference documents with every window and accounts for every page.

Parameter (evaluated configuration)	Value
Window size	At most 40 pages and 8 MB. A document of up to 50 pages is reviewed in one call
Request ceiling	24 MB on the wire, below the API's 32 MB limit
References	A long reference with a text layer is sent whole as page-tagged text; a long scanned reference is sent as its first 50 pages, with the gap declared

Parameter (evaluated configuration)	Value
Page resolution	Native PDF pages at full resolution. Fallback only: a single page too large to send as PDF is sent as a 1,024 × 1,024 image
Output budget	32,000 tokens per window, shared with reasoning
Retries	Four attempts per window, transient errors only

Windowing

Windows are packed by the measured size of each page, because page sizes on scanned records vary widely (13 KB to 1.9 MB on one record). Every page goes into exactly one window. A request that is still too large is split in two at full resolution, and only a single page that cannot fit on its own is sent as an image. Each window's header states its page range.

Caching

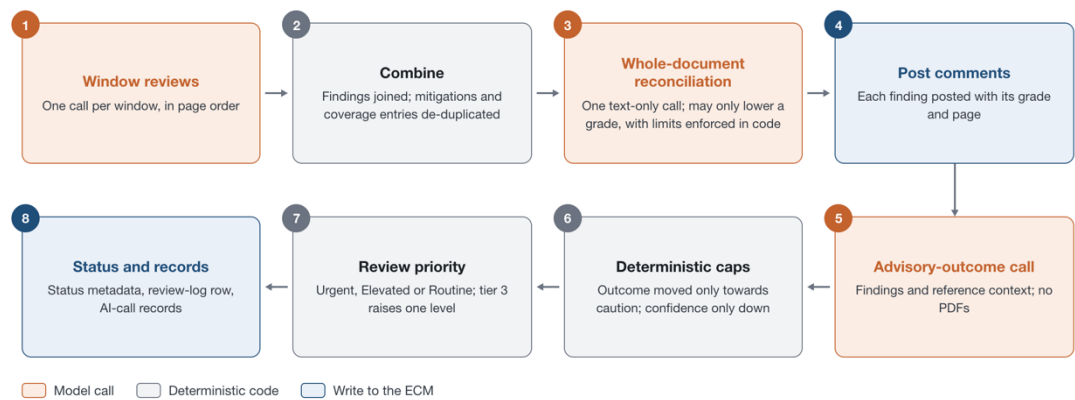
The system prompt and the references are cached, so windows after the first read them at a fraction of the price. In the evaluated pass, caching cut the AI model-call cost by an estimated 57%.

Coverage ledger

For each review, CDP records which pages the AI model read, which it did not and what it declined to evaluate. The ledger keeps the invariants below.

Invariant	How it is kept
Every page is in exactly one window	By construction of the window planner, and checked in tests. In the evaluated pass, each of 1,061 pages went to the AI model in exactly one of 38 windows
Every window ends read, or unread with a reason	Failed, refused and cut-off windows are recorded as unread with the reason, never as zero findings (window answers are not checked against the schema; Supplementary Note 11)
Unread pages are declared on the document	An "INCOMPLETE REVIEW COVERAGE" notice names the pages and says whether a re-run will recover them
Partly attached references are declared	An "INCOMPLETE REFERENCE COVERAGE" notice names the document and the pages not sent
Content shown but not evaluated is declared	Chart pages the AI model reports it did not read against a limit are gathered into a "TREND CHARTS NOT EVALUATED" notice. The notice depends on the AI model's own report
A window's source is stamped by code	Never taken from the AI model
Coverage notices always reach a reviewer	They bypass the switch that quiets other comments. They are posted in one attempt, and a failed post is logged
Nothing is posted when every page was read	Avoids a confirmation comment on every review

7.5 Reconciliation, grades and the advisory outcome



Supplementary Fig. 5 | Position of reconciliation in the pipeline. Reconciliation runs once over the whole document after the window reviews, and before comments are posted and the advisory outcome is set. Deterministic caps are applied in code after the advisory-outcome call. Orange, AI model calls; grey, deterministic code; blue, writes to the ECM.

Reconciliation

A window that raises a yield excursion may never see the approved deviation recorded 150 pages later. Reconciliation is the one step that sees the whole document. It is a single text-only AI model call that receives every finding, mitigation and coverage entry and returns an adjusted grade for each finding, with the evidence for any change. It runs before comments are posted (Supplementary Fig. 5). Its limits are enforced in code as well as in the prompt:

- A grade can only be lowered, never raised, and no finding is deleted.
- A blocker lowered because of a recorded mitigation falls below major only if the mitigation is approved or closed. A major can be lowered on any mitigation recorded in the document.
- A finding can be refuted by coverage only from a different window, only if the finding has an absolute page, and a blocker never falls below major this way. (In the campaign-report mode, coverage from a source document can lower a blocker below major.)
- On any error, the windows’ grades stand.

The reviewer sees the original grade and the reason. In the evaluated pass, 7 of the 10 findings first graded major were lowered, each citing a mitigation recorded in the record.

Grades

Findings carry one of four triage grades.

Grade	Meaning
Blocker	Would prevent disposition on its own, for example an out-of-specification result or an obliterated recorded value
Major	A real compliance or data-integrity concern to resolve before the record is complete
Minor	Genuine, with low impact
Informational	Context

Other severity words the AI model may use are mapped to these four in code (for example, critical to blocker), and any unknown value is stored as major. The grade orders the reviewer's queue. It is not a deviation class, which only the quality unit assigns after investigation.

Advisory outcome

One AI model call takes the findings and the reference context, without the PDFs, and returns Recommend Release, Needs Investigation, Recommend Reject or Insufficient Data, with a confidence of High, Medium or Low and a one-sentence rationale. Deterministic caps are then applied in code. They can move the outcome only towards caution and the confidence only downwards.

Condition	Effect
Any blocker after reconciliation	Recommend Reject
A review category failed or returned nothing that could be read (single-call reviews only; in windowed reviews, unread pages are reported by the coverage notice)	Insufficient Data at Low confidence
A reference came from the product fallback	Confidence Low
A cited method was mapped to a pharmacopoeial chapter	Confidence Low
A required reference is missing	Confidence Medium
A reference was picked by the AI model	Confidence Medium
A finding mentions an out-of-specification result	Confidence Medium
The document had no text layer	Confidence Medium
The MBR found does not match the batch record's vendor identifier	Confidence Medium
A cited specification is missing and the fallback lands on a different document series	A blocker is added, forcing Recommend Reject

Insufficient Data outranks Recommend Reject, so a review that cannot reach a conclusion forces a human look. The caps that fired are recorded in the review log and the AI-call log.

Review priority

Priority is Urgent for Recommend Reject or any blocker, Elevated for any major and Routine otherwise. It is raised one level for a tier-3 (highest-risk) document. The tier is a weighted score of context signals, such as a first document from the vendor, the first of its type for the program and material, or an unresolved required reference, and tier 3 needs a score of 5 or more.

7.6 Re-review and comment preservation

Delta re-check

When a vendor uploads a revised version, a person can request a delta re-check. One AI model call (one per window for long documents) receives the revised PDF, the prior reviewed version (in single-request mode only, when it fits), and up to 30 open comments. It returns a verdict for each comment (addressed, partially addressed, unresolved or not applicable), any new blocker and a short summary. CDP posts each verdict as a reply under the comment, with one roll-up comment. In windowed mode,

positive evidence from any window wins, and "unresolved" is used only if every window was read; otherwise the verdict is "unconfirmed: check the unread pages by hand". Comments people deleted are treated as rejected and are never raised again.

Full re-review

Before a full re-review, CDP:

1. Stops if a person has commented or replied on the file, and posts a note suggesting the delta re-check instead. Nothing is deleted.
2. Snapshots every comment (identifier, author, time, text and file version) to the comment-audit ledger. The snapshot is best effort, flushed hourly, and a failed snapshot does not stop the next step.
3. Deletes only its own earlier comments. A person's comment is never deleted.

All three controls passed their criteria in the configuration register, and the first was verified with a second, human account. Comments are still removed in two other ways: a repeated delta re-check replaces CDP's own earlier delta comments, unless a person has replied, and a reviewer who dismisses a comment deletes it, because the ECM has no "resolved" state.

Supplementary Note 8 | Data integrity and the audit trail

8.1 Records CDP keeps

Record	Content
AI-call log	One record for every AI model call in review, extraction and the registries, refused calls included (fields below)
Review log	One row per review category: status, the references and how each was found, the number of findings, the advisory outcome and confidence, the risk tier and the AI model
Comment-audit ledger	A snapshot of every comment before any deletion, and snapshots at review time and at send-back
Ingest failures and worker health	Why a file was held; health files for each worker

AI-call log fields

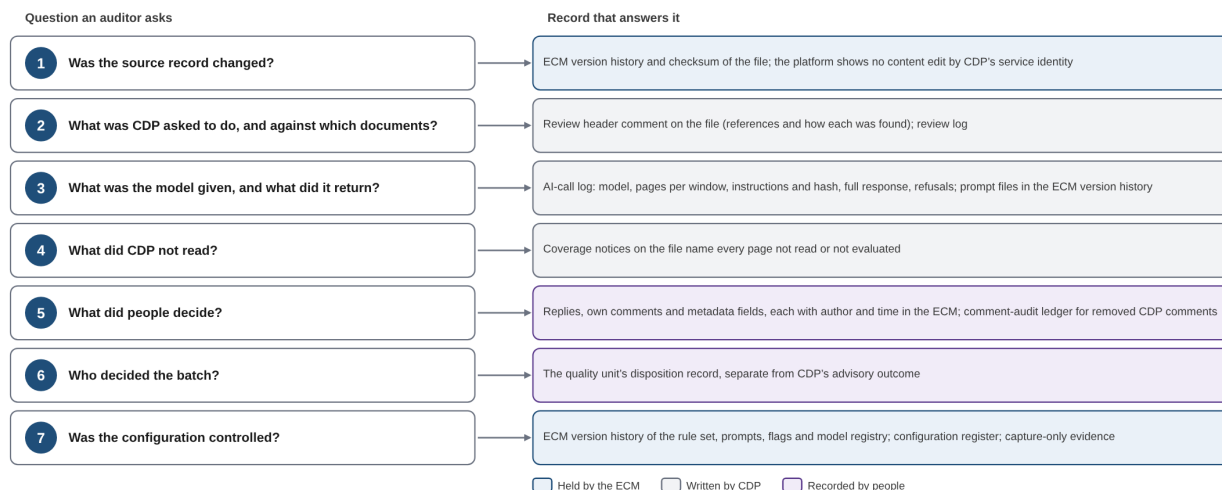
Field	Content
Run identifier, time (UTC)	Unique identifier; ISO-8601 time stamp
File identifier and name, document type	The evaluated document
Category	The review category and window
AI model, effort	The AI model that answered, and the reasoning effort
Prompt hash, length and text	In a single-call review, the full prompt. In a windowed review, the window's own header and user text only; the shared system prompt is not stored with the record, and the version in force is found from the ECM's version history
Attached files	For each file: its role (for example "Reference: Master Batch Record") and identifier
Response	The AI model's full text
Token use	Input, output, cache-read and cache-write tokens
Caps, risk tier, error	The confidence caps that fired, the risk tier and any error

8.2 How records are written

Records are appended to a local daily file on the worker's persistent volume, and that day's file is copied to CDP's protected area in the ECM every hour. A record that cannot be written locally is set aside as a self-describing envelope, which an operator writes back with a replay script that is safe to run more than once.

8.3 Reconstructing one review

Starting from one reviewed batch record, a reviewer, an auditor or an inspector can reconstruct the review from the records in the ECM and the audit log (Supplementary Fig. 6).



Supplementary Fig. 6 | Reconstructing one review. Each question an auditor would ask about a review, and the record that answers it. Blue records are held by the ECM, grey records are written by CDP and purple records are made by people under their own sign-in.

8.4 Data-integrity attributes

Every human action carries the person's ECM identity, and every machine write is attributed to CDP's service account. ECM time stamps, and the UTC time stamps and run identifiers in the logs, record when each action happened. Source documents are left unchanged, so the original is always kept. The identifier registry and the confirmation queue support accuracy, and the coverage ledger supports completeness.

Supplementary Note 9 | Security and access

Our Quality function's controls specification sets the security and data-handling requirements. The table gives, for each control, how the installed system meets it.

Control	How it is met
No write path to source documents	The AI model has no tools and no credentials. The application writes only metadata, comments and audit records, and never edits document content; renaming, moving and re-versioning are switched off. The application's ECM identity has read-write scope, because writing metadata and comments needs it, so the limit is enforced in application code and configuration
Machine output is additive	A re-review never deletes a person's comment. CDP's own earlier comments are snapshotted to the comment-audit ledger before they are removed (best effort)
Service identity, no impersonation	At run time the application signs in as its own service account and never acts as a user. Three set-up commands can act as an administrator, only when one is configured for that run
A named person decides	No code path sets an approval or a rejection. CDP has no user interface of its own: people act in the ECM under their own sign-in, so authentication and attribution come from the ECM
Untrusted document content	Vendor documents are treated as untrusted input. The AI model has no tools, credentials or network access, so text inside a document cannot make the system act; it can only affect the advisory comments a person reviews
Encryption	All external traffic is outbound TLS on port 443. The hosting platforms encrypt data at rest
Secrets	One vault per environment, read through the workload's managed identity at start-up. None are stored in code
Isolation	The workers expose no inbound endpoint and connect outbound only, to the ECM, the AI model API and the vault. Sandbox and production are separate tenants and clusters
Segregation from other validated systems	CDP connects to one validated repository, the ECM. It has no connection to other GxP systems
Least privilege	Configuration and audit areas sit in a sponsor-managed part of the ECM that vendors cannot see, restricted to system owners
Data minimization	Each request carries only the document or window, its references, extracted metadata and the instructions. Documents are sent within each request and are not uploaded for storage. The provider's standard data-retention terms apply, with no zero-data-retention agreement
Monitoring	Calls counted by endpoint in each cycle, health files, a watchdog, and monitors of production health and document activity. Dependency updates are proposed automatically
Environment safety	The environment selector, a red production banner and the start-up tenant check (Supplementary Note 4.3)
Fail visibly	Dependency checks, a frozen event cursor, review errors that force human review, coverage notices and comments on held files

Supplementary Note 10 | Operational controls

10.1 How people use CDP

CDP automates logging and classifying documents, finding the references each must agree with, and writing up graded findings. People start the review, decide on each finding, and send the document back or accept it, in the ECM, using CDP's comments and a small set of metadata fields (main-text Fig. 1b).

#	Step	Who
1	Log the incoming document	CDP
2	Classify and extract (files stay where they were uploaded)	CDP
3	Record identifiers, lots and genealogy	CDP
4	Start the review	A person, with one click
5	Find the review reference documents	CDP
6	Review against the references	CDP
7	Post graded comments and the advisory outcome	CDP
8	Decide on each finding: agree, dismiss or add	A person
9	Request a send-back	A person, with one click
10	Draft the email to the vendor from the open comments	CDP drafts; a person edits and sends it. Nothing is sent automatically
11	Request a delta re-check of the revised version	A person, with one click
12	Delta re-check	CDP
13	Accept, or send back again	A person, outside CDP

Reviewer actions

A reviewer can dismiss a finding (by deleting the comment), add their own comment, request a send-back or re-review, correct metadata, confirm a flagged identifier, reclassify a document, and add or exclude references. A type set by a person is never overwritten.

Feedback

A comment-delta audit compares comments at review time with comments at send-back and sorts them into deleted, added by a person and retained. This ledger is the record of what reviewers kept, rejected and added. Prompt changes made from it follow the configuration management of Supplementary Note 5.

10.2 Capture-only mode

A review run in capture-only mode makes every AI model call for real but writes nothing to the ECM except the end-of-review comment snapshot: no comments, deletions, status fields, review-log rows or AI-call records. Token use is recorded for each call so that the run can be priced.

10.3 Evaluation before activation

AI model changes, and flags that can change review output, are evaluated in capture-only runs before they are switched on. Capture-only mode covers the review path only, so an ingestion test used a write guard in the test script that blocked every write. Code changes are tested by the automated suite (Supplementary Note 10.4).

10.4 Tests and test evidence

CDP has about 3,000 automated tests, run on every push. They include a static check of call signatures across all modules, integration tests that run the real window loop with only the ECM and the AI model faked, and seam tests checked by mutation. A GxP test-evidence report runs the suite and renders a document with the code version, the source of the flag values, the configuration state, pass and fail tables for each suite, and a cross-reference from each feature to its tests. An end-to-end test follows a document from a review request through the pipeline and the review to a send-back, and is re-run after major changes. Prompt and AI model changes are evaluated by repeated runs on a fixed set of records with known findings.

Supplementary Note 11 | Known limitations

11.1 Components not in the evaluated configuration

Component	State
Routing and renaming of files into a standard folder tree	Built; switched off in every environment. Files are classified and extracted where they were uploaded
Email intake	Built; switched off
Review on upload, and automatic delta review on a vendor's re-upload	Built; switched off. Reviews run when a person requests them
Second-model check of linking identifiers	Applies only to the ECM's built-in AI extraction, which is disabled, so it does not run
Trend analysis in review	Switched off

11.2 Practical limitations

Reviewers and quality staff should know the following when they use CDP's output.

Limitation	What it means in practice
Variation between runs	The same record can give different findings, grades and advisory outcomes on a re-run. The grade is the least reproducible output, so the reviewer decides how serious each finding is
Some rules suppress real findings	Signatures are left to the reviewer because records reach CDP before the vendor's sign-off, and documented corrections are treated as complete. Issues in these classes are not raised and stay on the reviewer's checklist
Chart pages	Trend-chart pages the AI model did not read against a limit are declared in a notice, but they do not change the advisory outcome. A reviewer must read them
Review reference documents	For executed batch records the MBR is optional, so an unresolved MBR does not lower confidence. Upstream lot records are not linked, so charge quantities cannot be traced to the record that produced them. When several equal candidates match, CDP labels the pick [ambiguous] but does not lower confidence
Page numbers	Page anchors are converted to absolute pages in code, but in some windows a page number can be misread as absolute and the anchor misplaced. The text of a comment raised in a window can also quote the page number within that window
Unchecked answers	Window answers are not checked against the output schema, so an empty or unreadable answer would be stored as zero findings. None occurred in the evaluated pass
Audit trail	In windowed reviews, the AI-call record holds the window's own instructions but not the shared system prompt, which must be recovered from the ECM's version history
Duplicates	Findings are not de-duplicated across windows, so the same issue can appear twice
Dismissals	A reviewer dismisses a finding by deleting the comment, because the ECM has no resolved state. No reason is recorded
Revised review reference documents	A revised specification or MBR does not trigger a new review, and an upload that arrives before its reference is not retried automatically. A person must request the re-review
Instructions inside documents	Text in a vendor document could alter or suppress findings, and no defense against this has been tested. The AI model has no tools, so such text cannot make the system act; the risk is an omission from an advisory output

Limitation	What it means in practice
One AI model provider	CDP depends on one AI model provider, with no fallback to another provider's AI models

11.3 Silent failures found and fixed

Supplementary Table 3 lists the failures that occurred without any visible signal during development and operation, and the fix for each.

Supplementary Table 3 | Failure modes observed, mitigations and status

#	Class	Failure mode	Consequence observed	Mitigation	Status
1	Silent partial read	A failed page fetch (rate limit, server error or timeout) was treated as the end of the document	A record of about 80 pages was reviewed as 51 pages, and an outcome was posted with nothing to say so	Only a not-found response ends a page walk. Back-off that honours retry-after. Page count read first. Single download and local slicing. Coverage notice	Fixed
2	Silent partial read	An authentication expiry partway through a page walk was read as the end of the document	A campaign-report review was reconciled against 7 of its 9 source records without notice	Token refreshed on expiry, with the header shared across the walk	Fixed
3	Silent partial read	A window reported as missing data that sat in another window	A false major finding for missing hourly in-process checks, reported by a process owner	Coverage addendum. Window sources stamped in code. Downgrade-only reconciliation across windows	Fixed
4	Silent partial read	Extraction read only the first 50 pages (classification, 30) of oversized documents	List fields that accumulate across pages came out short without warning	Chunked extraction up to 400 pages, with list fields combined across windows	Partly mitigated. Beyond 400 pages the chunked path stops and writes a log line
5	Silent partial read	A reference-reconciliation pass ran outside the retry wrapper, with its warning switched off	An overload error dropped a source record from a summary review without notice	Retry added. Notice made visible to reviewers	Fixed
6	Request size	Raw bytes compared with a ceiling meant for bytes on the wire	Repeated permanent request-too-large errors; one document failed five times	Separate ceilings for raw and wire bytes. Packing by measured page bytes. Permanent errors reported as permanent, with the pages named	Fixed

#	Class	Failure mode	Consequence observed	Mitigation	Status
7	Request size	Windows sized by the average page size	A slice 4.2 times the ceiling, on a record whose page sizes varied 27-fold	Packing by measured page bytes. Splits at full resolution	Fixed
8	Request size	The PDF rendition of an office file never arrives	A large slide deck was never classified or reviewed, and the failure was logged as an oversized file	Two tries, then stop. The uploader is told and the file is excluded	Fixed
9	AI model behavior	Refusals on a few specific inputs	Every window of a long record was refused when a specification that triggered refusals was attached in error. Fallback answers were cut off at the output budget of 32,000 tokens per window	Refusal fallback. The answering AI model recorded for each window. A raw mode for evaluation that disables the fallback	Fallback in place. Issue open
10	AI model behavior	Run-to-run variation (sampling cannot be set, and there is no seed)	Few findings appeared in every one of three identical runs, and repeat findings were often re-graded	A person dispositions every comment. Severity is treated as the reviewer's decision. Consistency is measured for each change	Open; measured for each change
11	AI model behavior	Placeholders on blank forms read as recorded deviations	A form legend was reported as a mitigation	A prompt prohibition, and a code check that rejects placeholder patterns	Fixed
12	Prompts and rules	No de-duplication in code	Three of the 45 comments in the evaluated pass were adjudicated duplicates	A rule to report each issue once, and a consolidation rule	Partly mitigated
13	Prompts and rules	Families of comments needing no action that the rules required (rounded totals, handwritten ranges)	27 of the 33 no-action comments in the evaluated pass	Rounding now judged against the decimal places of the limit	Rule changed; effect on the scored records not measured
14	Page anchoring	The conversion of window-relative pages to absolute pages is ambiguous when a window starts at or before its own length. Comment bodies keep window-relative page numbers	In the evaluated pass, 21 of 45 comment bodies quoted window-relative pages	Window page labels. Conversion to the absolute page in code. Out-of-range pages clamped to the document	Anchor conversion in place; comment bodies not rewritten

#	Class	Failure mode	Consequence observed	Mitigation	Status
15	Reference resolution	A template number shared between batches, with the ECM's first result labelled exact	A batch record was reviewed against the MBR of a different batch	Tie-break on batch identity. Larger, ordered candidate pools. An [ambiguous] label. Reviewer-added and excluded references	Fixed
16	Reference resolution	A fallback list cut short before filtering, with no dosage-form dimension	Records were reviewed against a specification for the wrong dosage form	Declarative candidate matching. The reference is left unresolved when ambiguous	Fixed
17	Reference resolution	Identifiers invented by the ECM's built-in AI extraction	Protocol numbers that were not on the page	Identifier registry built only from the documents that define identifiers, with a confirmation queue	Fixed
18	Reference resolution	No re-review when a review reference document is revised; early uploads are not retried	Stale or missing references persist	Manual re-trigger. Diagnostics name the fix	Deferred by design
19	Cost	AI model-API credit ran out during a run	An overnight comparison run was voided	Manual monitoring	Procedural
20	Cost	A mismatch in status vocabulary between two components caused a re-review loop	Repeated reviews of the same item	Vocabulary aligned. Integration test	Fixed
21	Scope	The write guard blocked both the failure record and the comment on a held file	Held files carried no comment explaining the hold	Guard narrowed to the document tree. The comment always posts	Fixed
22	Scope	The routing step ran while file moves were off	Spurious holds on files	With routing off, the routing step does not run	Fixed
23	Scope	Deleting a comment also deletes the replies to it, and comments have no parent pointer or resolved state	People's replies could be removed with the CDP comment they answered	No deletion if a person has commented. Snapshot before deletion	Fixed
24	Configuration	Configuration pushed ahead of the code that reads it	Workers ignored unknown keys without warning	Pre-push checklist. Per-tenant status report	Procedural
25	Configuration	The prompt seed drifted from the live prompts, and some prompt text was compiled into code	Repository prompts differed from the live prompts, and hidden text contradicted new rules	Live configuration is authoritative. Prompt text moved to configuration. A check on AI model identifiers in code	Fixed

#	Class	Failure mode	Consequence observed	Mitigation	Status
26	Testing	A half-applied scripted edit passed 2,536 tests that replaced the called function with a stub	Every windowed re-review broke on the development cluster	Static contract test of call signatures. Integration tests of the real window loop	Fixed
27	Testing	The capture-only setting was not passed into worker threads	A comparison run wrote 16 audit records that capture-only mode should have suppressed	Setting copied into threads. Regression test	Fixed
28	Operations	Local recovery files depend on a persistent volume	None observed	Persistent volume on the event worker. Hourly copy to the ECM. Flush on graceful shutdown	In place

Supplementary Note 12 | Mapping to regulatory expectations

Table 2 of the main text summarizes the mapping by principle. Supplementary Table 4 maps each control to the expectation it serves and to the objective evidence that can be examined. The expectations come from 21 CFR 211¹¹ and Part 11¹², EU GMP Chapter 4¹⁴ and Annex 11¹⁶, draft Annex 22¹⁵, ICH Q7¹⁰, Q9(R1)²⁶ and Q10²⁷, GAMP 5²⁸, FDA's draft guidance on AI²⁹ and data-integrity guidance^{17,30,31}. The table below sorts the controls by how each is enforced.

Enforced by	Controls
Application code, with tests	No write to document content. Routing, the phase whose code can move or re-version a file, switched off by configuration. Page citation in the output schema. Routes for uncertainty, through prompt rules and deterministic caps. The audit-log writer. Capture-only mode. Caps on the advisory outcome. Downgrade-only reconciliation
The hosting platform or infrastructure	Segregation from other validated systems. Encryption. Network isolation. Role-based access and individual sign-in. Monitoring. Retention settings on the audit-log location
Written procedures and records	Quality-unit disposition. Promotion to production. Capture-only evaluation of AI model changes and of configuration that affects output. Configuration management of prompts, rules and AI models. Routing of concerns about the AI through the quality system

Supplementary Table 4 | GMP expectations, CDP controls and the evidence an inspector could examine

#	Expectation (instrument)	Control in CDP	Evidence
1	The quality unit reviews batch records and decides each batch, and the duty is not delegated (21 CFR 211.22, 211.192; ICH Q7 §2.22, §6.71; draft Annex 22 §1)	Comments and an advisory outcome only. Reviewers disposition each comment, and the quality unit decides the batch	Written procedures, signed disposition records, platform audit trail of send-back requests and metadata changes
2	Records are changed only by authorized people, without obscuring the original (21 CFR 211.68(b), 11.10(e); EU GMP Ch. 4 §4.9; Annex 11 §9)	No write path to the source document. Comments and metadata only. Renaming and moving disabled	File version history before and after review, audit log of the service identity, unit tests
3	Secure, time-stamped audit trails that can be reviewed (21 CFR 11.10(e); Annex 11 §9; MHRA §6.13; ICH Q7 §5.43)	Review, extraction and registry AI model calls logged with the AI model, response, token use and instructions (shared prompts identified by file version) in an append-style log in the system of record	Call-log files, retention setting, steps to reconstruct a review (Supplementary Note 8.3)
4	Attributable actions and authority checks (21 CFR 11.10(d), (g), 11.50; Annex 11 §12, §14)	Individual sign-in for every human action. Configuration folders kept separate from review folders. The application acts under its own service identity	Identity and role configuration, access records, platform audit trail
5	Computer output is checked for accuracy (21 CFR 211.68(b); EMA reflection paper ³² §2.3.5)	Each comment is anchored to its page and cites its review reference document. The review header shows how each reference was resolved	Comments, review header, disposition of each comment

#	Expectation (instrument)	Control in CDP	Evidence
6	Uncertain outputs are flagged (draft Annex 22 §9.2; ICH Q9(R1) §5.1)	Unconfirmed findings say so. Insufficient Data outranks Recommend Reject. Product-level fallbacks and MBR identifier mismatches cap confidence	Versioned prompts, cap rules and their tests
7	The whole record is reviewed (21 CFR 211.192; ICH Q7 §6.70–6.72)	Each page sent in exactly one window. Unread pages and content not evaluated are declared in notices with page anchors	Coverage notices, window-level call logs
8	Computerized systems are validated to a depth set by risk (21 CFR 211.68, 11.10(a); ICH Q7 §5.40–5.41; Annex 11 §1, §4; GAMP 5)	Used as an advisory aid and not validated as a GMP computerized system. Installation-qualification register. About 3,000 automated tests, run on every push	Advisory-use risk classification, installation-qualification register, test-evidence report
9	Changes, including configuration, are controlled (Annex 11 §10; ICH Q7 §5.47; ICH Q10 §3.2.3; draft Annex 22 §10)	Versioned rules and prompts with an owner. Prompt version and AI model recorded for each review. Sandbox or capture-only evaluation before AI model and configuration changes; quality sign-off for AI model changes	Configuration version history, installation-qualification register, versions in review headers and logs, capture-only reports
10	No loss of quality assurance when a system replaces part of a manual operation (Annex 11 Principle; draft Annex 22 §4.3, §5–6)	Locked manual baselines, an auditor-standard rubric and a planned held-out validation with a pre-defined protocol	Protocol and analysis plan, adjudication sheets
11	Human review of AI output is recorded and monitored (draft Annex 22 §3.3, §10.5; ICH Q10 §4.1)	Comment snapshots record which findings reviewers kept or removed (best effort; dismissals are captured at send-back). Reviewer feedback feeds prompt revision	Comment-snapshot ledger, consistency reports, training records
12	Third parties that process data are under agreement and oversight (Annex 11 §3; ICH Q10 §2.7)	Each request carries only the document, its references and the instructions. In the evaluated configuration, documents are sent within each request and are not uploaded for storage. Standard provider retention, with no zero-data-retention agreement	Supplier assessment, data-processing agreement, data-flow map
13	Security and incident management (Annex 11 §12–13; 21 CFR 11.10(d); ICH Q7 §5.46)	Encryption in transit and at rest. A secrets vault for each environment. Separate sandbox and production	Infrastructure definitions, vault policies, corrective and preventive action records

Appendices

Appendix A | Output schema

Appendix Table 1 | Output schema fields

Field	Type	Required	Set by	Rule
findings[]	array	Yes (may be empty)	AI model	Zero findings is a valid answer
findings[].title	string	Yes	AI model	5–10 words, the headline of the concern
findings[].severity	enum	Yes	AI model; reconciliation may lower it	blocker, major, minor or informational. Other words are mapped in code
findings[].body	string	Yes	AI model	One sentence of at most 30 words, with the page, section and value inline. Non-English text is quoted with an English gloss
findings[].page	integer	When page-anchored	AI model; converted to the absolute page in code on windowed reviews	PDF page counted from the first page shown. Never guessed. Omitted for findings about the whole document
findings[].region	[x0, y0, x1, y1]	No	AI model	Bounding box in fractions of the page, when the finding can be localized
findings[].references	array of role names	Yes	AI model	1–3 role names, for example "Master Batch Record"
mitigations[]	array	No	AI model	Recorded deviations, justifications, corrections and change controls seen in the document
mitigations[].type	enum	Yes	AI model	deviation, justification, correction, change_control
mitigations[].ref	string	Yes	AI model; checked in code when reconciliation cites it	A real recorded identifier. A placeholder from a blank form is rejected
mitigations[].pages	string	Yes	AI model	PDF page or range
mitigations[].summary	string	Yes	AI model	At most 20 words
mitigations[].status	enum	Yes	AI model	approved, closed, open, referenced, unknown
coverage[]	array	Windowed reviews	AI model	What this window documents, specific enough to contradict a false claim that something is absent
coverage[].documented	string	Yes	AI model	Item, value or time span, and page

Field	Type	Required	Set by	Rule
coverage[].pages	string	Yes	AI model	Pages within the window
coverage[].source	string	—	Code only	"pages S–E". Never accepted from the AI model
charts[]	array	Windowed reviews	AI model	Pages whose plotted values were not read against a limit. Not counted as findings
charts[].what, charts[].pages	string	Yes	AI model	Description and pages
Advisory outcome, confidence	—	Never returned by the review call	Separate call, then code	One of four outcomes with High, Medium or Low confidence; caps applied in code
Original grade and reconciliation note	—	—	Reconciliation call, limits enforced in code	The original grade and one clause citing the mitigation or coverage and its page
Version tag, category tag, page marker	—	—	Code	Added when the comment is posted

Appendix B | Redacted prompt template

The template below paraphrases the structure and intent of the common system prompt and the executed-batch-record prompt. Every element specific to a program, vendor or language has been removed, and it is not the production text. The implementing organization fills in the fields in square brackets. The worked-example slot is synthetic by design.

SYSTEM ROLE

You are [SYSTEM NAME], an advisory reviewer in a pharmaceutical CMC document pipeline. You help Quality by identifying discrepancies, gaps and points of concern in the document under review by comparing it with the reference documents provided.

Rules

1. Advisory only. A person makes every release decision. Your findings become comments.
2. Return the JSON object defined in OUTPUT only.
3. Cite evidence inline: page, section or step, and the value exactly as printed. Refer to documents by role name ("Master Batch Record"), never by file identifier.
4. Each finding must be understandable from its title and body alone.
5. Zero findings is a valid answer. Do not invent concerns.
6. Quote numeric values exactly, with units and significant figures.
7. A missing reference is a finding only when the document itself cites something that is absent or contradictory. Never raise a finding because the pipeline did not supply a document. Never name internal metadata fields.
8. The PDF is ground truth. Extracted metadata is supplementary; if it disagrees with the PDF, trust the PDF and move on.
9. Write findings in English. When citing non-English text, quote the original and add an English gloss. Bilingualism is not a finding; disagreement between language versions of a value, criterion or date is.
10. When one class of issue recurs, raise one consolidated finding naming the pattern and representative locations, but only when the occurrences are themselves findings.
11. Cite the absolute PDF page shown in each page's label; add the printed page in parentheses where it differs. If unsure, cite the step and omit the page. A missing page is acceptable; a wrong one is not.
12. Signatures, initials, signing blocks and the dating of signatures are out of scope. Still in scope: the document's own effective dates as a chronology anchor, the two-person check on critical recorded values, and the integrity of recorded data. CDP reviews executed records before the vendor's quality sign-off, when signatures are often not yet complete.

13. Report mitigations (recorded deviations, justifications, corrections, change controls) separately, with their real identifiers, pages and status. Do not suppress your own findings because of them. Never report a blank-form placeholder.

14. [REVIEW-RULE FAMILIES: see Supplementary Note 7.3]

Severity (triage grade, not a deviation class)

blocker: would prevent disposition on its own.

major: real compliance or data-integrity concern to resolve before the record is complete.

minor: genuine, low impact.

informational: context; use sparingly.

DOCUMENT TYPE: EXECUTED BATCH RECORD

Scope: one consolidated pass with two lenses. Report every issue once.

- Quality lens: good documentation practice and ALCOA+, handwritten entries, deviation cross-references, completeness of required fields.
- Technical lens: process parameters against their ranges, equipment, in-process controls against criteria, yield and material reconciliation, deviation impact.

Escalate on uncertainty: a potential excursion you cannot confirm or rule out from the record is raised and phrased "unconfirmed: reviewer to verify" (blank required field in an executed step: blocker; unconfirmed potential excursion: major; a value inside its limit is not a finding).

Branch-driven sections: first establish which branch the batch took. Blank entries inside a branch that was properly marked not applicable are expected and are not findings. Raise one finding only if the routing decision is unrecorded, the wrong branch was executed or marked not applicable, or the marking is incomplete.

Handwritten entries: legibility; corrections that leave the original legible, with reason; "see deviation" notes must point to a recorded deviation; impossible chronology. Obliteration of a recorded value is the highest grade.

Derived values: where the record gives inputs and the governing reference gives a rate, duration or ratio, compute it and compare. "The limit is not printed beside the value" exempts the layout, never the value.

Context: the executed record normally carries its own limits; the MBR and specification, when provided, are enrichment. Do not raise findings for their absence.

NOT findings: absence of a separate MBR or specification; bilingualism itself; the adequacy of a deviation's root-cause analysis; cross-batch trends; MBR template design.

[SPONSOR CONVENTIONS: rounding convention, correction-reason codes, branch markings]

[WORKED EXAMPLES: 3–6 synthetic before/after pairs, one per high-volume review-rule family (in-limit value; total against components under the stated rounding convention; documented correction; not-applicable branch), each showing the expected finding or the expected silence. Never take examples from records in the evaluation or held-out partitions.]

OUTPUT

[Common output schema; Appendix Table 1]

(windowed reviews only) COVERAGE ADDENDUM

You are seeing one slice: pages S–E of T. Never assert that anything is absent, missing or never performed anywhere in the record. Describe only the pages in front of you. Emit coverage[] entries naming what these pages document (item, value or time span, page) and charts[] entries for pages whose plotted values you did not read against a limit. Do not set a source field; the framework stamps the window.

Appendix C | AI Model calls

Appendix Table 2 | AI model calls in the ingestion and review paths

In the evaluated configuration, review calls ran on Claude Opus 4.7, with Claude Sonnet 4.6 configured as the refusal fallback (not invoked in the pass). Calls made through the router fall back on a refusal, a rate-limit error or a server error; window review and reconciliation fall back on a refusal only. Review, extraction and registry calls each write one audit record (Supplementary Note 8).

Call	Trigger	Inputs	Output	On refusal, failure or invalid output
Classification	Upload, or a request to reclassify	The opening pages of the document, with the classifier prompt	Document type (27 types) and material category (6)	Fallback AI model on refusal. A type set by a person is never overwritten. A failed classification holds the file with a comment
Extraction	After classification	Template fields for the document type, with field-level instructions. Windows of up to 400 pages, with list fields combined	One value per field. List fields as plain strings	Three attempts, then the file is held with a comment
Row-level registries	After extraction	The PDF	One row per process parameter, result, time point or lot, with a confidence for each row	Reviewers triage rows below high confidence. A failure in genealogy never blocks processing
Reference disambiguator	Several equal candidates for one reference	The evaluated document and each candidate	The index of the pick, or abstain; a confidence; a rationale	On abstention or any failure, the deterministic pick, labelled [ambiguous]
Pharmacopoeial backstop	Method citations still unresolved after every other step	At most 50 pages of the evaluated document, with the citations	For each citation, a standard or none, with a confidence and a rationale	No answer leaves the reference unresolved
Window review	A review request	Cached system prompt, references, metadata block and one window	Appendix Table 1	On refusal, one fallback, after which the window is unread. Output budget 32,000 tokens per window
Reconciliation	After all windows, when mitigations or coverage entries exist	Findings, mitigations and coverage entries, as text only	For each finding, an adjusted grade and the evidence cited	Fallback on refusal only. Any error leaves the windows' grades unchanged
Advisory outcome	After reconciliation	Findings text and the reference context, with no PDFs	One of four outcomes, a confidence and a one-sentence rationale	No fallback. An answer that cannot be parsed is treated as Insufficient Data at Low confidence. Caps applied in code
Delta re-check	A delta request	The revised PDF, the prior version (single-request mode only) and the open comments	A verdict for each open comment, any new blocker and a summary	Verdicts merged across windows. "Unresolved" only when every window was read
Send-back draft	A send-back request	The remaining comments and the file's context	Subject, body and number of issues	Draft only. Nothing is sent automatically
Trend narrative	A trend detected in code (certificate and stability types)	The trend's data points	One sentence without judgement words	No fallback