CONTROLLED DOCUMENT

Validation Guide — CMS Life Sciences Sample Management System

DOCUMENT	Reference
SUMMARY	Supplier validation guide. Technical description of the Sample Management system's GxP controls, how it is built and released, the division of responsibility for validation activities, and the sample protocol templates supplied with it.
OWNER	CMS Life Sciences
CREATED	2026-09-09
UPDATED	2026-09-09

Validation Guide

CMS Life Sciences Sample Management System

Document ID	CMS-VG-001
Version	1.0
Effective date	*(issue date)*
Supersedes	—

Revision history

Version	Date	Change
1.0	*(issue date)*	First issue

1. Purpose and scope

This guide describes the GxP controls implemented in the CMS Life Sciences Sample Management system, the process by which the system is built and released, and the supplier documentation supplied with it.

It is written for a validation or quality function planning and executing the qualification of an installed instance. It covers the system as delivered. Configuration applied during implementation is described in the Configuration Specification issued for that instance.

1.1 Standards referenced

Reference	Application
21 CFR Part 11	Electronic records — controls, audit trail, record retention and retrieval
EU GMP Annex 11	Computerized systems
ICH E6 (R3)	Good Clinical Practice, where samples are collected under a clinical protocol
ISPE GAMP 5 (2nd ed.)	Software categorization, lifecycle and supplier documentation
MHRA GxP Data Integrity Guidance	ALCOA+ record attributes
ISO 20387	Biobanking

1.2 Software categorization

The system is a configured product — **GAMP 5 Category 4**. The application is commercially supplied and common to all installations. Study configuration, metadata schemas, workflow routing, role definitions, storage topology, retention policy and label templates are configuration held as data, applied per instance, and are not code changes.

No customer-specific code is written into the core application. Where an installation requires behavior the configuration model does not express, that behavior is delivered as a separately identified extension and is categorized, specified and tested as Category 5.

---

2. System identification and architecture

2.1 Deployment

The application is deployed as a versioned container image into the customer's own cloud tenancy, under the customer's identity provider. Application data, audit records and attachments reside in storage owned by the customer and are not replicated to supplier-controlled infrastructure.

The supplier holds no operational access to a production instance. Support access, where granted, is issued through the customer's identity provider under the customer's own access controls and is subject to the same audit trail as any other session.

2.2 Components

Component	Function
Application runtime	Serves the operator surfaces and the API; enforces authorization at every route
Governance layer	Records every state change as an audit event; assigns actor and time server-side
Record store	Holds sample, custody, shipment, exception, consent and configuration records
Attachment store	Holds files linked to records, with metadata held in the record store
Reporting surfaces	Compose stored records into operator, custodian and auditor views and exports

2.3 Identity

Authentication is delegated to the customer's identity provider over Azure AD / OpenID Connect. The application does not store or verify passwords, and issues no credential of its own. Session subjects originate from the identity provider and are the basis of all attribution.

---

3. The GxP record model

The following are GxP records in this system. Each is subject to the audit trail described in section 4 and the retention controls in section 8.

Record	Content
Subject	Donor or subject identity reference, and the consent records governing use
Consent	Consent instrument, permitted uses, effective period, linked documents
Sample	Sample identity, type, study assignment, metadata, status, current location
Aliquot	Sample derived from a parent, carrying its own identity and lineage relationship
Container	Physical container identity, including externally supplied barcodes
Shipment / parcel	Inbound and outbound consignments, contents, condition and temperature evidence
Custody event	Transfer of holder or location, with acting party and time
Exception / defect	Discrepancy raised against material or a manifest, with lifecycle and disposition
Distribution	Request, approval, dispatch, return and reconciliation of material
Disposition	Final outcome of a sample, with authorization and reason
Configuration	Study, workflow, role, retention and label configuration
Audit event	The record of every change to any of the above

Sample identity and lineage, location, custody, condition and disposition are mastered in this system. Study design and the study master file are external master data, consumed by reference. Analytical results are mastered by the analytical system and referenced rather than held.

---

4. Audit trail

4.1 Generation

Every write that changes a GxP record produces exactly one audit event. Event generation is a property of the governance layer through which all writes pass; it is not a call that application code may omit. A write that produces no corresponding audit event fails the build.

4.2 Event content

Field	Content
Event identity	Unique identifier for the event
Record identity	The record changed, and its type
Actor	The authenticated subject responsible for the change
Timestamp	Time of the change, applied server-side in UTC
Action	The operation performed
Reason category	The structured reason for change (section 6)
Comment	Free-text justification where required
Prior value	Each changed field's value before the change
New value	Each changed field's value after the change
Record attribute	The data integrity attribute the event evidences (section 7)

Changes are recorded field by field. A single operation affecting several fields produces one event carrying the individual before-and-after pair for each field changed, so that the specific difference is legible rather than inferred from two record states.

4.3 Attribution

The acting subject is stamped server-side from the authenticated session. An actor identifier supplied by a caller is discarded and not recorded. A caller therefore cannot attribute a change to a subject other than the one authenticated, and attribution cannot be forged through the API.

Changes not originating from a human action — scheduled evaluations, automated status transitions, unattended inbound processing — are attributed to the system or to the named functional account responsible, and are distinguishable from changes made by a person.

There is no shared or generic operator account. Every session resolves to an individual subject in the customer's identity provider.

4.4 Time

Timestamps are applied by the server at the moment the change is committed. A client cannot set, offset or backdate the time of an event. Times are stored in UTC and rendered in the viewer's configured timezone, with the stored value unchanged.

4.5 Immutability

The audit trail is append-only. The store exposes no update or delete operation for an audit event, and no application path writes one. A correction to a record is recorded as a further event carrying its own reason and prior value; the

superseded value remains readable.

Audit records are retained independently of the records they describe. Deletion of an attachment, or disposition of a sample, does not remove the audit history of either.

4.6 Review and export

Audit events are readable through an audit trail viewer, filterable by record, record type, subject, date range, action and reason category. Field-level changes render as a before-and-after comparison.

Events are exportable as CSV for a selected scope. The export is generated from the stored events and carries the same fields as the viewer.

Audit trail review is supported as a recorded activity: a review is itself a record, capturing the scope reviewed, the reviewer, the period covered and the outcome.

---

5. Attribution and access control

5.1 Authorization model

Access is granted by role. A role carries a set of permissions; permissions gate routes, actions and fields. Authorization is evaluated on every request at the route boundary, independently of any check performed by a handler, so a route added without its own check is refused rather than exposed.

A subject holding no permission for a surface is refused with the permission required named, rather than shown an empty or partial view.

5.2 Field and record scope

Permissions apply at three levels: which functions a subject may perform, which records a subject may reach, and which fields within a record a subject may read or change. Study, site and business-group scoping restrict record visibility to the subjects entitled to it.

5.3 Privileged actions

Actions with elevated consequence — configuration change, retention policy change, bulk update, waiver of an exception, restoration of a deleted attachment, disposition authorization — are individually permissioned and separately auditable from ordinary operational writes.

Where an action requires a second person, the approving subject must differ from the subject who initiated it, and the system refuses an approval by the initiator.

5.4 Access review

Current role assignments, the permissions each role carries, and the last authenticated session for each subject are reportable for periodic access review. Removal of a subject's access is recorded as a governed change.

---

6. Reason for change

A change to a committed record requires a reason before the change is accepted. The reason is selected from a structured vocabulary rather than typed freely:

Category	Application
Correction of error	Corrects a data-entry or process mistake
Additional information	Adds detail not available at the time of entry
Protocol amendment	Reflects a change to the study protocol
Sponsor request	Made at the sponsor's direction
Regulatory requirement	Required to meet a regulatory obligation
System-generated	Automated correction, attributable to the process that made it
Other	Requires a free-text justification

The category is stored on the audit event and is filterable. A comment is mandatory on `Other`, on waiver and resolution of an exception, on bulk update, and on deletion of an attachment.

Bulk operations carry one recorded reason for the operation while writing an individual audit event against each affected record, so that a bulk change is as traceable and as reversible as a single one.

---

7. Data integrity

Each record attribute is enforced by a mechanism rather than by procedure. Every governed write is stamped with the attribute it evidences, and the stamped vocabulary is closed — an attribute with no enforcing mechanism renders as unmapped rather than being absent from the register.

Attribute	Mechanism
Attributable	The acting subject is stamped server-side from the authenticated session and cannot be supplied by the caller (4.3)
Legible	Records are stored as structured data and rendered from it; coded values resolve to their configured labels in every surface and export
Contemporaneous	The timestamp is applied at commit by the server and cannot be set by the client (4.4)
Original	The first committed value is retained on the audit trail; a later value never overwrites the record of the earlier one (4.5)
Accurate	Field constraints, controlled vocabularies and referential checks are enforced at the write; an invalid value is refused rather than stored and flagged
Complete	Every governed write produces an audit event, and a write path producing none fails the build (4.1)
Consistent	Sequence is established by server-applied time and by the append-only event order; related records share one vocabulary and one identity scheme
Enduring	Records and their audit history persist in customer-owned storage independently of the application instance, and survive its replacement
Available	Records and audit history are retrievable through the operator surfaces, the reporting surfaces and the export paths throughout the retention period, without recourse to the supplier

8. Retention, archival and disposition

Retention periods are configuration, set per study or per record class, and are themselves governed records. A change to retention policy is audited with its prior value.

An archived study is locked: records within it become read-only, and transitions the retention standard forbids are refused with the governing rule named. The configuration governing an archive renders with the source it derives from.

Deletion of an attachment is permission-gated and requires a comment. The file is held in a soft-deleted state and does not appear in file lists; its metadata and audit entry persist, and a system administrator can restore it within the retention window. The record that something was attached, and that it was removed, outlives the attachment.

Sample disposition is a governed and authorized act, recorded with the acting subject, the authorization, the reason and the time. A disposition record stands independently of the sample record it terminates, and is readable against the consent and retention schedule that governed it for as long as the retention period requires.

9. Retrieval for inspection

Records and audit history are retrievable by the customer without supplier involvement, through the operator surfaces, the reporting surfaces and CSV export.

The following compose stored records into inspection-facing views:

Report	Content
Chain of custody	Audit events, distribution and disposition composed into one timeline per sample, from receipt to final disposition
Audit trail	Filterable event history with field-level before-and-after comparison
Ethics and consent	Consent instruments in force, the samples they govern, and re-use decisions taken against them
Data integrity pack	The record attributes of section 7, with the enforcing mechanism and the evidence for each
Access review	Role assignments, permissions carried, and last authenticated session per subject
Retention register	Records approaching or exceeding their retention period, and the disposition applied

Exports are generated from the stored records at the time of export, and carry the identity of the exporting subject and the time of generation.

---

10. Configuration

10.1 What is configuration

Study definitions and their metadata schemas; workflow routing and approval paths; role definitions and the permissions they carry; storage topology and capacity rules; placement and quarantine rules; label templates and barcode formats; retention periods; exception lifecycles and reason vocabularies; notification rules; and interface field mappings.

Configuration is held as data, is versioned, and is readable as a single artifact. A quality function reads the definition governing an installed instance directly rather than reading a description of it.

10.2 Configuration change control

A configuration change is a governed write. It requires the corresponding permission, is recorded on the audit trail with its prior and new value, and — where it affects a validated behavior — is subject to the approval routing configured for it.

Configuration is promoted between environments rather than edited in place, so that the definition qualified in a lower environment is the definition that reaches production.

10.3 What is not configuration

Changes to the application's own behavior, data model or control mechanisms are code changes, delivered through the release process in section 12, and are not made in a running instance.

---

11. Interfaces

Inbound and outbound interfaces are declared as configuration: the source or destination, the transport, the field mapping to and from the internal model, and the validation applied on receipt.

Inbound records are validated against the required-field set before acceptance. A record failing validation is rejected with the failing field named and the rejection recorded; it is not partially applied. Re-ingestion of a corrected source merges onto the existing records against the configured external identifier, without creating duplicates, and every field-level change is audited with its prior value.

Interface activity is recorded: what was received or sent, when, from or to which system, the volume, and the outcome including rejections.

---

12. Development lifecycle and change control

12.1 Change origination

Every change enters as a tracked unit of work carrying the requirement it satisfies. A change without a stated requirement has no path into a release.

12.2 Verification

Requirement coverage is enforced mechanically. Automated tests bind requirements to the behavior that satisfies them, and the build fails where a requirement's tests do not pass or where a governed write is added without its audit event. Verification is a condition of the build rather than a reviewer's recollection.

Defects are recorded against the requirement they violate, so that defect history is readable per requirement rather than per release.

12.3 Release and versioning

Versioning is explicit and monotonic. Environments are promoted rather than edited, so the artifact qualified is the artifact deployed. Every artifact is built from a versioned source revision, and the revision, the build inputs and a checksum are recorded with it, so an artifact can be reproduced from retained inputs and reconciled against its source.

Release documentation — what changed, against which requirements, what it was tested against, who approved it — is generated by the release pipeline from the change record.

12.4 Change affecting a validated state

A change affecting a validated behavior is delivered as a prepared package: the impact assessed against the installed configuration, the affected requirements identified, and regression evidence supplied. The package enters the customer's change control. Deployment into the customer's environment is performed by the customer.

---

13. Supplier testing evidence

The following is generated by the build and supplied for each release.

Evidence	Content
Requirement traceability	Each requirement mapped to the specification that defines it and the tests that exercise it, with requirements carrying no test named rather than omitted
Functional test results	Per-requirement assertions with their results, versioned against the release they ran on
Recorded demonstrations	Screen recordings of the tested behavior, addressable per requirement
Audit trail integrity evidence	Verification that governed writes produce their corresponding events and that the trail is unbroken
Defect record	Defects raised against the release, their disposition, and the requirements affected
Release documentation	Change content, approvals, test basis and version identity

This evidence establishes that the system behaves as built and as specified. It is generated by the same test specifications that gate the build, so the evidence supplied and the verification performed are the same activity.

---

14. Division of responsibility

R — performs. A — accountable for the outcome.

Activity	Supplier	Customer
Validation plan, scope and risk assessment	—	R / A
User Requirements Specification	—	R / A
Supplier assessment or audit	Supports	R / A
Functional, Configuration and Design Specifications	R	A
Requirement traceability	R	A
IQ and OQ protocols	Templates supplied	R / A
IQ and OQ execution	—	R / A
PQ and user acceptance testing	—	R / A
Deviation investigation	Supports, against supplier code	R / A
Validation summary report and release for GxP use	—	R / A
Periodic review and audit trail review	—	R / A
Change impact assessment and regression evidence	R	A
Deployment into the customer environment	—	R / A
Backup, restore and continuity testing	—	R / A
Infrastructure qualification	—	R / A

The supplier performs no validation activity in the customer's environment and holds no executed qualification evidence.

Supplier personnel are available at a published hourly rate to answer questions on system behavior, explain implementation, and investigate against supplier code during a customer's validation.

---

15. Supplied documentation

Document	Form
This guide	Issued per release
Functional Specification	Issued per release
Configuration Specification	Issued per instance
Requirement traceability matrix	Generated per release
Test evidence and recorded demonstrations	Generated per release
Release documentation	Generated per release
Sample IQ protocol	Template — Appendix A
Sample OQ protocol	Template — Appendix B
User guide, administrator guide, configuration guide	Issued per release

Sample protocols are supplied at no cost. They are derived from the supplier's own release testing and are written against a baseline configuration. A protocol as supplied is not evidence; evidence is produced by its execution in the installed environment.

Appendix A — Sample Installation Qualification protocol

(template)

Appendix B — Sample Operational Qualification protocol

(template)

Appendix C — Requirement traceability matrix

(generated per release)