IntuitionLabs

INTUITIONLABS / RESEARCH & PRACTICAL GUIDANCE

AccessGUDID API v3: Build a Versioned Device Master

> Build practical AI for pharma and biotech with IntuitionLabs. We help life-science teams turn complex information and workflows into useful software, governed knowledge systems and AI tools.

Published by IntuitionLabs · 2026-09-28 · accessgudid api v3 · accessgudid · gudid · medical device master · unique device identifier · device identifier · public device record key · medical device data · gmdn · snomed ct

Original article: https://intuitionlabs.ai/articles/accessgudid-api-v3-versioned-device-master

In this article

Executive Summary

AccessGUDID API v3 is a public lookup and enrichment interface for device records in the US Global Unique Device Identification Database (GUDID). It is useful for resolving a scanned device identifier (DI), a full unique device identifier (UDI), or a Public Device Record Key to public record attributes. It is not a bulk replication interface: the National Library of Medicine (NLM) also offers daily change files and monthly full releases. A governed master should use the **Public Device Record Key** as its durable record identity, retain every observed public version, and treat DIs as effective-dated identifiers because the primary DI can change while the record key remains stable. (accessgudid.nlm.nih.gov)

The distinction matters operationally. A UDI contains a fixed DI and, when present on the label, variable production identifiers (PIs) such as lot or serial number. GUDID stores the DI portion, not those PI values. A scanned full UDI therefore belongs in a controlled event or inventory layer; the public device master holds its parsed DI and public attributes. Separate tables for record versions, identifiers, package relationships, terminology, and source provenance prevent a DI change or packaging revision from silently overwriting history. (www.fda.gov) (hl7.org) (www.hhs.gov)

As of **September 28, 2026**, NLM listed **5,205,424 DI records** in its September full release, compared with **5,182,695** in August, a net snapshot difference of **22,729** records. The listed September 28 daily file contained **6,130** DI records. Those figures are release inventory counts, not a count of newly introduced products or a claim about current market supply. The full release supplies a reproducible baseline; daily files maintain it; history lookup resolves a specific record's public versions; a filtered download is useful for a bounded cohort. The search interface's **10,000-record** limit is not a limit on full downloads. (accessgudid.nlm.nih.gov)

The practical design is to ingest a complete release, verify its published checksum and row count, apply change files in order, then reconcile the resulting current projection against the next full release. API v3 adds GMDN code, code status, and implantable terminology information to lookup and history; SNOMED enrichment has a distinct Unified Medical Language System (UMLS) authentication boundary. No AccessGUDID-specific request rate limit was verified in the opened official documentation, so clients should

back off on errors and avoid treating a different API's limit as this one's. The resulting master remains **public device identification information**, with separate manufacturer-controlled product data and regulatory assessment. (documentation.uts.nlm.nih.gov) (www.fda.gov) (www.fda.gov)

Introduction and Background

A hospital materials system, laboratory catalog, or medtech regulatory warehouse often starts with a barcode scan and a question: which model does this code identify, and which public facts were true when the transaction occurred? A flat table keyed only by the scanned string cannot answer that reliably. The full UDI may contain a lot, serial number, expiration date, or manufacturing date. Its DI identifies the labeler and the version or model. FDA says GUDID contains DI data but not PI values, so a public catalog and a physical-unit record have different scopes. (www.hhs.gov)

GUDID is the FDA's repository of submitted device identification information. **AccessGUDID** is NLM's public search, application programming interface (API), and download surface for that information. FDA says the public service can be used without an account, while publication follows a seven-day editing grace period. The public feed is therefore an observable public state rather than a real-time view of a manufacturer's internal product master. (www.fda.gov) (www.fda.gov) (catalog.data.gov)

This report treats a device master as a set of linked, versioned facts, not one row per barcode. It covers API v3 documentation and endpoint choice; the DI, full UDI, record key, and package identifier hierarchy; release and history ingestion; controlled terminology; data-quality tests; and downstream use in inventory, clinical, and laboratory systems. The engineering guidance is a proposed implementation pattern derived from the cited interface behavior, not an NLM reference architecture. NLM's own pipe-delimited files provide a relational view for bulk import, and health IT guidance expressly allows current GUDID attributes to be served from a local copy. (healthit.gov)

For an adjacent consultancy perspective, IntuitionLabs describes data engineering and business intelligence services and an information layer built around authoritative enterprise sources, identity, permissions, and citations. Those capabilities explain why a locally governed public-data layer may be useful; they do not make the consultancy a device database or a source of GUDID record truth. (intuitionlabs.ai) (intuitionlabs.ai)

Identity and Versioning: What the Master Must Represent

Separate the identifier layers

A **full UDI** is the encoded label value, while the **DI** is its mandatory fixed model-level component. The **PI** is conditional and variable. The DI can be used to reach a public GUDID record, but the full UDI may identify a particular production lot or physical item. HHS distinguishes the model-level DI from serial and other values tied to a particular device when discussing de-identified data. Keeping those values in separate domains supports both correct joins and appropriate handling of patient-linked events. (www.fda.gov) (www.fda.gov) (www.hhs.gov)

- **Public Device Record Key:** Use as the durable public record key. NLM says it does not change for an individual record even if the primary DI changes.

- **Primary DI:** Store as an effective-dated identifier relationship, not the warehouse primary key. API lookup accepts DI and record key as separate inputs.
- **Secondary and unit-of-use identifiers:** Preserve identifier type, issuing agency, and relationship to the primary device record; the lookup response contains an identifiers collection. (ref.gs1.org)
- **Package DI:** Model each package identifier with quantity and package status as a child relationship. GS1's healthcare rules distinguish trade-item packaging levels. (ref.gs1.org)
- **Full UDI and PI:** Parse for events, inventory, or patient records, then protect lot and serial values according to that system's rules. GUDID itself does not store PI values. (www.hhs.gov)

NLM added public version number, version date, status, and record key in **April 2018**. A historical version can be **new**, **update**, or **delete**. For records already present before that change, public version 1 is implied; do not invent a precise pre-2018 event history. A deleted record is no longer searchable through the ordinary interface or APIs, which is a reason to retain snapshots and release provenance in the local master. (accessgudid.nlm.nih.gov)

An identity example

Consider a hypothetical device record with key `R`, primary DI `D1`, and a later public version naming `D2` as primary. The identity table keeps one record `R`; the identifier table closes the current interval for `D1` and opens one for `D2`; the version table keeps both public versions. This is a **hypothetical example**, not a claim that any named device made that transition. The pattern follows NLM's explicit statement that the record key survives a primary-DI change.

A package DI needs a separate edge to the model-level record rather than being flattened into a comma-separated field. It can have a different trade-item level and quantity. This separation matters when an inventory scan uses a case code but a clinical record needs the unit-level device description. GS1's standard assigns distinct identifiers to distinct trade-item packaging levels, while FHIR Device distinguishes the fixed DI from a full human-readable UDI carrier. (ref.gs1.org) (hl7.org)

Choosing the AccessGUDID Interface

The first interface decision is driven by **cardinality and purpose**. One DI or record key calls for lookup; one record's changes call for history; a scanned label calls for parse; terminology enrichment calls for SNOMED; a cohort calls for filtered download or a release file. *Table 1* compares the supported paths.

Path	Best input and output	Use in a versioned master	Boundary
v3 device lookup	DI, full UDI, or record key to one public record. (accessgudid.nlm.nih.gov)	Resolve exceptions and verify a current record.	A UDI input can also yield parsed information in response headers.
v3 history	DI, UDI, or record key to public versions. (accessgudid.nlm.nih.gov)	Reconstruct changes for an affected record.	Compare public versions for the same record key.

Path	Best input and output	Use in a versioned master	Boundary
v3 parse UDI	Full UDI to structured DI and PI components. (accessgudid.nlm.nih.gov)	Parse a scan before joining it to master data.	PI values belong to the transaction or unit context. (www.hhs.gov)
v3 SNOMED and implant list	Device terminology or a paginated implantable subset. (accessgudid.nlm.nih.gov) (accessgudid.nlm.nih.gov)	Add a separately sourced terminology layer.	SNOMED calls have UMLS authentication requirements. (documentation.uts.nlm.nih.gov)
Filtered or full release	Bounded CSV cohort or monthly full XML and delimited files. (accessgudid.nlm.nih.gov)	Bootstrap and reconcile a large local copy.	Search export's 10,000-result cap does not define the full-release size.

The matrix points to a hybrid pattern: release files for completeness, API calls for targeted resolution. A system that repeatedly pages through a list to build the entire device universe should first consider the published full release. Conversely, a single clinical scan should not wait for a full-file refresh when a lookup can resolve the current public record. NLM's full release is monthly and its daily files carry changes received during business days. (healthit.gov)

Endpoint behavior and authentication

The v3 paths documented by NLM are `/api/v3/devices/lookup`, `/api/v3/devices/history`, `/api/v3/parse_udi`, `/api/v3/devices/snomed`, and `/api/v3/devices/implantable/list`, with documented JSON or XML variants. Lookup accepts `di`, `udi`, or `record_key`; the UDI should be percent-encoded. Its parsed UDI headers appear only for a UDI input, so a client must preserve both body and relevant headers.

- **Lookup:** Prefer a record-key query after local identity resolution. Save the returned public version metadata and raw response for audit.
- **Device History API:** Request by record key when possible. The history page lists `record_key`, `di`, and `udi`, although its introductory parameter sentence is narrower; test the chosen parameter against the live endpoint.
- **Parse:** Supply the percent-encoded full UDI. NLM documents GS1, HIBCC, and ICCBBA formats.
- **SNOMED:** Treat terminology as an optional enrichment with its own authentication path. The endpoint page specifies a UMLS single-use ticket.
- **Implant list:** Use the documented page and date parameters, and persist navigation and count headers for a reproducible crawl.

NLM announced in **March 2023** that Device SNOMED and Implantable List calls could accept a UMLS API key in addition to ticket workflows, while the live v3 endpoint descriptions still emphasize tickets. Implementations should confirm the exact current v3 query parameter with a real authorized request before relying on the API-key option. The UMLS REST API's published **20 requests per second per IP** term is for that UMLS service, not a verified AccessGUDID v3 rate limit. (accessgudid.nlm.nih.gov) (documentation.uts.nlm.nih.gov) (documentation.uts.nlm.nih.gov)

Canonical Schema and Change-Aware Ingestion

A robust schema preserves the **source record**, the **public version**, and the **business interpretation** separately. The public record key can anchor the first two. A canonical identifier table should record DI value, type, issuing agency, package role, observed version interval, and source release. A terminology table should hold GMDN, SNOMED CT, and FDA product-code assertions independently, because their provenance and update rules differ. *Table 2* gives a compact field dictionary. (www.gmdnagency.org)

Canonical object	Key fields and source	Change rule
Public record	`record_key`, first-seen release, raw payload pointer.	Stable warehouse identity; retain provenance.
Public version	`record_key`, version number, date, status, source snapshot.	Append on new, update, or delete; do not overwrite.
Identifier edge	DI, type, issuing agency, package quantity and status from lookup.	Open or close an effective interval as relationships change.
Device attributes	Brand, model, company, distribution, MRI, sterility and other public fields.	Snapshot per public version; preserve nulls and source wording.
Classification	FDA product code and GMDN code, status, term from public data. (www.fda.gov)	Version each assertion; do not merge code systems.
SNOMED enrichment	Concept identifier, terminology version, retrieval time. (www.snomed.org)	Refresh independently under the applicable licence.
Transactional UDI	Full carrier, parsed DI, lot or serial, event time in a controlled system. (www.hhs.gov)	Keep outside the public model-level master.

The table makes a common migration error visible: if a team overwrites a row keyed by primary DI, a changed DI can look like a new device with no lineage. The record key and public version table preserve lineage. A changed brand, distribution status, or MRI statement then becomes a version event rather than an unexplained discrepancy. NLM's lookup output carries distribution status, product classification, and MRI-related fields; the schema page defines the public version structure.

Bootstrap, daily updates, and reconciliation

A practical load starts from a named **monthly full release**. Retain its timestamp, file name, published checksum, and raw file hash, then pin the published schema artifact and its version history. This keeps parser behavior reproducible when a future download has different fields.

1. **Land immutably:** Store downloaded parts, hashes, source URL, retrieval time, and release label.
2. **Validate transport:** Compare the published checksum, part count, archive integrity, and record count before parsing.
3. **Normalize records:** Parse identifiers and repeating children into separate tables; retain an unmodified source representation.
4. **Apply change files:** Process daily files in release order and interpret public status as an event, including deletes.

5. **Resolve exceptions:** Query v3 lookup or history for an ambiguous record key or DI transition.
6. **Reconcile monthly:** Compare the current projection to the next full release by record key and version, then investigate any mismatch.

NLM says a daily release is supplied every business day and a weekly release bundles recent daily files. Treat the weekly package as a convenience or recovery route, not as extra additive changes on top of the same daily files. The monthly full release is a new independent baseline. Because FDA describes a seven-day editing grace period before public appearance, a missing new submission cannot be interpreted as a permanent absence. (accessgudid.nlm.nih.gov) (www.fda.gov)

Terminology, Attributes, and Downstream Trust

FDA product codes, **Global Medical Device Nomenclature (GMDN)** terms, and **SNOMED CT** concepts answer different questions. A product code is a regulatory classification field in the public record. GMDN is a nomenclature code and term for a device type; the GMDN Agency defines its code as five digits and governs database use under licence terms. The SNOMED endpoint returns associated clinical terminology, but it has a separate UMLS authentication dependency. Keep each code system, version, status, and source as distinct assertions. (www.gmdnagency.org)

NLM's **2023 v3** change added GMDN code, code status, and an implantable flag to lookup and history outputs. (accessgudid.nlm.nih.gov) A prior integration that only loaded term names loses information needed to distinguish a current code from a retired one. SNOMED International's map access requires GMDN licence confirmation (www.gmdnagency.org), and in **January 2026** the two organizations announced a planned extension to replace an existing linkage table. That announcement is a reason to version mapping provenance, not to presume the replacement is already deployed in AccessGUDID. (www.snomed.org) (www.snomed.org)

- **Description:** Keep labeler-supplied brand, model, catalog number, and free text beside normalized terms; do not let one overwrite another.
- **Distribution:** Retain the public commercial-distribution status and its version date; it is a public statement, not live inventory. (www.fda.gov)
- **MRI and sterility:** Preserve original categorical values and nulls. Do not infer an attribute from a related device or package.
- **Package hierarchy:** Use an explicit parent-child edge and quantity; the package code can be what a scanner sees. (ref.gs1.org) (hl7.org)
- **Cross-jurisdiction terms:** Store European Medical Device Nomenclature (EMDN) separately if a global catalog needs it; the European Commission says EMDN is used for EUDAMED registration and is freely available. (health.ec.europa.eu)

The trust boundary is as important as the schema. A GMDN code does not by itself determine whether an item meets a jurisdiction's medical-device definition. AccessGUDID exposes public identification data, while FDA maintains a separate database of cleared and approved device information. The manufacturer-controlled item

master, regulatory assessment, purchasing status, and patient or inventory events need their own owners and evidence. An AccessGUDID match should enrich those systems without silently declaring that an item is approved, currently sold, safe, or stocked. (www.gmdnagency.org) (www.fda.gov) (www.fda.gov)

For hospitals, health IT certification guidance explicitly discusses recording an implantable device UDI, parsing it, and associating GUDID attributes. It also permits those attributes to come from a current local system. HL7 FHIR Device can represent the full UDI carrier, fixed DI, and lot number in distinct fields, while a catalog-level DeviceDefinition describes a kind of device rather than a physical instance. Those distinctions support a design in which the public master joins to patient or inventory events through DI but does not absorb their unit-level PI values. (healthit.gov) (hl7.org) (www.hhs.gov)

Data Analysis and Evidence

The current release inventory gives a useful sizing baseline, with limits on interpretation. NLM's listed **September 2026 full release** contained **5,205,424 DI records** in a **519 MB ZIP**; the listed **August 2026** release contained **5,182,695**. Subtraction gives a net difference of **22,729 records**, about **0.44%** of the August count. This calculation compares two published snapshots. It does not isolate additions, deletions, or changed versions, and a DI count is not a count of physical units or unique manufacturers. (accessgudid.nlm.nih.gov) (www.fda.gov)

The listed **September 27 weekly** file contained **71,577 DI records**, while the **September 28 daily** file contained **6,130**. (accessgudid.nlm.nih.gov) These counts represent records in those release files, which may include updates and removals; adding them to the full-release total would double count or misclassify changes. A warehouse should instead compare distinct public record keys and event statuses after ordered application. NLM says full files are monthly, weekly files bundle daily files, and daily files are issued on business days.

Table 3 turns the observed release metadata into controls. The formulas are proposed reader-side measurements, not published NLM performance benchmarks.

Control	Calculation or test	What an exception means
Transport integrity	Compare downloaded bytes with published MD5; retain a stronger local hash if policy requires.	File transfer or archive must be rechecked before loading.
Full-release count	Compare parsed DI rows and distinct record keys with the release's published DI count.	Parser, duplicate, or interpretation issue needs review.
Net snapshot difference	Current full count minus prior full count, using named monthly releases.	A net change does not identify additions or updates.
Change-file coverage	Count distinct released dates received divided by expected business-day dates in the chosen interval.	A missing file or schedule assumption needs review.
Freshness	Observation time minus latest successfully applied release date.	Stale downstream projection, even if API lookup is current.
Attribute completeness	Non-null valid values divided by applicable records, reported by field and public version.	A source-null or mapping change, not automatically a labeler error.

The table separates **integrity**, **coverage**, and **freshness**. A checksum can show that bytes match a published archive but cannot prove that a parser captured every child identifier. A row count can show population alignment but cannot establish accurate terminology joins. A fresh daily load can still lag a manufacturer's internal change because public availability follows the GUDID process. These controls should be reported together, with exceptions tied to release and record keys.

For bounded exports, the public search engine and full-record export have a **10,000-result** ceiling; NLM offers a filtered ZIP/CSV path for larger matching sets and points users to the full release when a needed filter is unavailable. This is a workflow limit, not a verified v3 lookup rate limit. The separate openFDA UDI endpoint documents a **1,000-record per-call** maximum, which must not be transplanted into AccessGUDID capacity planning. (accessgudid.nlm.nih.gov) (open.fda.gov)

Implications and Future Directions

A versioned master changes how teams answer seemingly simple questions. An inventory system can resolve a package scan to a model-level record while preserving package quantity. A laboratory catalog can resolve an in-vitro diagnostic (IVD) DI: Centers for Disease Control and Prevention guidance explicitly points laboratories to the manufacturer or AccessGUDID for that value. An electronic health record (EHR) can associate the public description with a recorded implantable UDI while retaining the actual scan and PI in the patient context. (ref.gs1.org) (www.cdc.gov) (healthit.gov)

Public identifiers also move into adjacent reporting ecosystems. The Centers for Medicare & Medicaid Services (CMS) publishes GUDID-derived device-name and primary-DI reference data for Open Payments and labels it reference data only. The Office of the National Coordinator for Health Information Technology broadened its UDI data element to include non-implantable devices in USCDI v6. Those uses increase the value of stable joins and recorded source dates, but they do not make every downstream representation identical to the current AccessGUDID record. (www.cms.gov) (healthit.gov) (hl7.org)

Five design decisions deserve explicit ownership:

- **Identity owner:** Decide whether an internal product key, public record key, and each DI are linked through a governed crosswalk. (www.fda.gov)
- **Source owner:** Record which fields come from GUDID, manufacturer systems, terminology licences, and local clinical events. (www.fda.gov) (www.snomed.org) (healthit.gov)
- **Change owner:** Define who reviews a changed primary DI, distribution status, package relationship, or classification mapping.
- **Privacy owner:** Keep lot and serial data in systems designed for their use; HHS distinguishes those PI values from model-level DI data. (www.hhs.gov)
- **Standards owner:** Pin terminology editions and revisit mappings when official releases change. The European Commission identifies EMDN releases by year, so a mapping should carry its terminology edition. (health.ec.europa.eu) (www.snomed.org)

As API and terminology documentation evolve, an integration should recheck endpoint contracts, schemas, and licence terms at each planned upgrade. This is especially relevant to the UMLS ticket/API-key discrepancy in the published NLM materials and to the announced GMDN-SNOMED mapping work. A small contract test

and a reproducible full-release reconciliation are more defensible than assuming a successful HTTP response means the device master is complete. (documentation.uts.nlm.nih.gov) (www.snomed.org)

Frequently Asked Questions (FAQs)

Is AccessGUDID API v3 sufficient to build the whole master?

Use it for targeted lookup, parsing, history, and enrichment. For a complete local baseline, use the monthly full release and then daily change files; NLM also supplies a filtered download for large bounded cohorts. Reconcile against a later full release.

Which key survives a primary DI change?

The **Public Device Record Key** identifies the public device record and remains unchanged for that record when the primary DI changes. Retain the old and new DIs as versioned identifier edges, and use public version number, date, and status to interpret the change. (accessgudid.nlm.nih.gov)

Does a full UDI lookup put lot or serial data into GUDID?

No. A full UDI input can be parsed, and lookup may return parsed information in headers, but FDA says GUDID stores only the DI portion. Keep PI values in the relevant transaction, inventory, or patient system. (www.fda.gov) (www.hhs.gov)

What are the pagination and request-rate limits?

The documented implantable list defaults to **100** results per page and allows **1 to 100**. Search and full-record export have a separate **10,000-result** ceiling. No AccessGUDID-specific request-rate number was verified in the opened official documentation; the published UMLS and openFDA limits belong to those other services. (accessgudid.nlm.nih.gov) (documentation.uts.nlm.nih.gov) (open.fda.gov)

Is SNOMED enrichment included in ordinary lookup?

Treat it as a separate endpoint or optional implant-list enrichment. The v3 SNOMED page specifies a UMLS single-use ticket; a March 2023 NLM notice says SNOMED API calls accept a UMLS API key. Confirm the live v3 authentication parameter before deploying it, and record terminology provenance.

Conclusion

A trustworthy **AccessGUDID v3 device master** starts with an identity model. The full UDI, fixed DI, public record key, package identifiers, public versions, terminology codes, and transactional PI values each have a different role. The record key anchors lineage; DIs and package codes are versioned relationships; public attributes are versioned assertions; lot and serial details remain with controlled events. (hl7.org)

The operational pattern is straightforward: bootstrap from a full release, verify its integrity and count, apply dated changes, use API v3 for targeted questions, and reconcile with the next full release. Keep endpoint authentication, code-system licences, schema artifacts, and freshness controls explicit. The public record is

valuable reference evidence, while manufacturer product truth, regulatory determinations, and local availability remain separate decisions.

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. https://accessgudid.nlm.nih.gov/help/download
2. https://www.fda.gov/medical-devices/unique-device-identification-system-udi-system/udi-basics
3. https://hl7.org/fhir/R4B/device-definitions.html
4. https://www.hhs.gov/hipaa/for-professionals/faq/can-device-identifier-di-portion-unique-device-identifier-udi-be-part-limited-or-de-identified/index.html
5. https://accessgudid.nlm.nih.gov/download
6. https://www.fda.gov/medical-devices/global-unique-device-identification-database-gudid/accessgudid-public
7. https://documentation.uts.nlm.nih.gov/rest/authentication.html
8. https://www.fda.gov/medical-devices/unique-device-identification-system-udi-system/global-unique-device-identification-database-gudid
9. https://www.fda.gov/medical-devices/device-advice-comprehensive-regulatory-assistance/medical-device-databases
10. https://www.fda.gov/medical-devices/unique-device-identification-system-udi-system/gudid-data-trends
11. https://catalog.data.gov/dataset/accessgudid
12. https://healthit.gov/test-method/implantable-device-list
13. https://intuitionlabs.ai/
14. https://ref.gs1.org/standards/gtin-allocation-healthcare/
15. https://accessgudid.nlm.nih.gov/download/schema
16. https://accessgudid.nlm.nih.gov/resources/developers/v3/device_lookup_api
17. https://accessgudid.nlm.nih.gov/resources/developers/v3/device_history_api
18. https://accessgudid.nlm.nih.gov/resources/developers/v3/parse_udi_api
19. https://accessgudid.nlm.nih.gov/resources/developers/v3/device_snomed_api
20. https://accessgudid.nlm.nih.gov/resources/developers/v3/implant_list_api
21. https://accessgudid.nlm.nih.gov/download/query
22. https://accessgudid.nlm.nih.gov/news
23. https://documentation.uts.nlm.nih.gov/terms-of-service.html
24. https://www.gmdnagency.org/licence-agreement/
25. https://www.snomed.org/maps
26. https://www.snomed.org/news/gmdn-agency-and-snomed-international-announce-strategic-collaboration-to-advance-global-medical-device-content-and-interoperability
27. https://health.ec.europa.eu/medical-devices-topics-interest/european-medical-devices-nomenclature-emdn_en
28. https://www.gmdnagency.org/faqs/
29. https://open.fda.gov/apis/device/udi/how-to-use-the-endpoint/

30. https://www.cdc.gov/locs/2020/guidance_for_livd_code_selection_forlabs_using_lab-dev-tests.html
31. https://www.cms.gov/priorities/key-initiatives/open-payments/resources/reporting-entitities
32. https://healthit.gov/standards-and-technology/onc-standards-bulletin/onc-standards-bulletin-2025-2
33. https://accessgudid.nlm.nih.gov/news-details?date=2023_03_01

THE PUBLISHER

About IntuitionLabs

IntuitionLabs is an AI consulting, custom software development and data engineering firm serving pharmaceutical, biotechnology, medical-device and other life-science organizations. We work with clinical, regulatory, medical-affairs, commercial, quality and IT teams to connect technology decisions with the work people need to accomplish.

AI consulting and adoption

Our AI enablement services cover readiness assessments, use-case selection, governance and policies, team workshops, adoption measurement and ongoing advisory support. We help organizations structure the information layer behind AI: source material, context, permissions and maintained knowledge that make generated answers useful and reviewable. Private LLM inference and hosted AI options support teams evaluating how to operate AI with appropriate control over their data and infrastructure.

Software, data and life-science workflows

IntuitionLabs develops custom software for pharma and biotech, integrates enterprise systems, and builds data engineering and business intelligence solutions. Areas of focus include AI agents, regulatory research, medical writing, medical affairs, CMC information, competitive intelligence and clinical-document workflows. Our eTMF intelligence work includes cross-system reconciliation and inspection-readiness support.

Enterprise platforms and regulated delivery

We provide Veeva services, application support, managed services, integrations and custom applications, alongside enterprise content work involving platforms such as Egnyte. For regulated workflows, our services include GxP enablement, computer-system validation and software development addressing 21 CFR Part 11 requirements. The applicable controls, validation responsibilities and acceptance criteria are defined for each engagement.

Work with IntuitionLabs

Explore AI enablement, pharma and biotech software development, data engineering and BI, and Veeva services. Contact IntuitionLabs to discuss your workflow, information sources and implementation needs.

IntuitionLabs publishes educational research to help life-science teams make informed technology decisions. Coverage of a product or organization does not imply a client relationship, endorsement or partnership.

Website: https://intuitionlabs.ai