Does Your LIMS Need to Be Validated? What Each Regulator Asks For

Five frameworks ask for the same thing in different vocabularies. None of them wants volume, and none of them lets a software vendor sign off for you.

Brandon Holland, Marketing & Operations Digital Solutions Architect

August 3, 2026 · 8 min read

IN BRIEF

As of August 2026, LIMS validation is owned by the lab, not the software vendor. If your records support an FDA-regulated product, an EU GMP submission, or an accredited testing scope, the system must be validated for your intended use, covering your configuration and workflows rather than the vendor's code.

Key takeaways

- Validation is documented evidence that the system does what your lab needs, in the configuration you actually run.
- The regulated party owns the validation. A vendor supplies evidence, protocols and people, never the signature.
- A hosted system changes the split of work, not the obligation: the vendor qualifies the platform, the lab its own use.
- Part 11, Annex 11, ISO 17025, GAMP 5 and FDA software assurance all ask for risk-based evidence, not paperwork volume.
- Validation is a state you maintain through change control, not a milestone you pass at go-live.

Somewhere between choosing a LIMS and going live, a question arrives that nobody put on the project plan: does this thing have to be validated, and by whom?

If your lab's records support an FDA-regulated product, an EU GMP submission, or an accredited testing scope, the system holding those records has to be validated for the way your lab actually uses it. Not certified. Not approved. Validated, by you, for your intended use. That distinction is where most of the confusion lives, and it is worth clearing up before a project starts rather than during an audit. This covers what validation actually includes, whether hosting changes it, who owns it, what each of the five frameworks asks for, and where the projects go wrong.

What does LIMS validation actually cover?

Validation is documented evidence that a system does what your lab needs it to do, consistently, in the configuration you are running.

The scope surprises people. It is not an inspection of the vendor's source code. It is an examination of your deployment.

- **Intended use.** A written statement of what the system is for in your lab. Everything else is measured against this, so a vague one makes the whole exercise vague.
- **Configuration.** Your specifications, your test methods, your workflows, your user roles. This is the part no vendor can validate for you, because it did not exist until you built it.
- **Data integrity controls.** Attribution, time stamps, the audit trail, retention, and who can change what. An audit trail that cannot be edited from the application is the difference between a record and a draft.
- **Interfaces.** Every instrument feed and every downstream system. A result that is transcribed by hand between two validated systems is an unvalidated step.
- **Change control.** What happens the next time you add a method or a role. Validation is a state you maintain, not a milestone you pass.

The vocabulary around this is inconsistent and worth pinning down. *Qualification* is the evidence for one component — the installed software, the operation of a function, the performance of a workflow — and *validation* is the conclusion drawn across all of it that the system is fit for its stated intended use. Installation, operational and performance qualification are the three classic layers; a vendor can supply the first and much of the second, and the third is yours.

Does a cloud LIMS still need to be validated?

Yes, and the split of work changes rather than the obligation.

With a hosted system, the vendor can qualify the infrastructure and the base platform once, and share that evidence with every customer. What the vendor cannot do is confirm that your stability study workflow behaves correctly, because that configuration is yours alone.

So the practical division is this: the vendor supplies platform and installation evidence, and the lab owns operational and performance qualification against its own intended use. Ask a prospective vendor what it hands over, in writing, before the contract rather than after. Lab management cloud hosting describes what the hosted arrangement includes; the validation obligation sits on top of it unchanged.

> NOTE
>
> A vendor that says its software “is validated” is describing its own development testing. Useful, and not the same thing as your lab’s validation. Ask what evidence you receive and what you are expected to produce.

Who owns the validation, you or the vendor?

The regulated party owns it. That is the lab, the manufacturer, or the accredited testing organisation, never the software company.

This is not a technicality that vendors hide behind. It follows from what validation is: a claim about fitness for a specific intended use that only the user of the system can make. A vendor can supply evidence, templates, qualification protocols and people who have done it before. The signature at the bottom is the lab’s.

It matters practically, not just legally. An inspector asking about a computerised system asks the lab what the system is for, who approved that statement, and what testing supports it. A vendor's certificate answers none of those three, which is why a lab that has outsourced the thinking rather than the paperwork tends to discover the gap in the room.

The useful division of labour is that the vendor owns everything that is true for every customer, and the lab owns everything that is true only for this one. Nobody else can tell you which of your workflows would matter if it failed.

What does each regulator actually ask for?

Five frameworks cover most laboratories, and they ask for the same thing in different vocabularies. Four are regulations or standards; the fifth is the industry guidance most validation projects are actually run to.

FRAMEWORK	APPLIES TO	WHAT IT ASKS FOR
21 CFR Part 11	Electronic records supporting FDA-regulated products	Validation of systems for accuracy, reliability and consistent intended performance, plus the ability to detect altered records
EudraLex Annex 11	EU GMP computerised systems	Risk-based validation, supplier assessment, documented change control, data integrity through the lifecycle
ISO/IEC 17025	Accredited testing and calibration labs	Information management systems validated for functionality before use, and after any change
FDA computer software assurance	Device production and quality management software	Risk-based assurance, with testing effort matched to the consequence of failure rather than to documentation volume
GAMP 5 (ISPE guidance)	Any regulated computerised system; the method most projects follow	A scalable lifecycle approach: categorise the software, assess risk, size the evidence to the risk, and lean on supplier activities where they are credible



Comparison of the five frameworks cited below, August 2026

Read across that table and the pattern is clear. None of them asks for volume. All of them ask you to know what the system is for, to test the parts where being wrong would matter, and to keep that evidence current as the system changes. GAMP 5 is worth naming separately because it is where the practical method comes from: its second edition in 2022 pushed explicitly toward critical thinking over templated documentation, which is the same direction FDA's software assurance guidance took. A pharmaceutical QC lab under Annex 11 and an accredited testing lab under 17025 are being asked the same three questions.

Where do validation projects go wrong?

Rarely in the testing. Almost always in the scoping.

WHERE THE EFFORT ACTUALLY LANDS

> **Before any testing**
>
> The intended use statement. Every hour spent after this is either aimed at it or wasted on something else.
>
> **During the project**
>
> Qualification against that statement, weighted to the workflows where being wrong would matter.
>
> **Every change after go-live**
>
> Change control. This is the half that decides whether the validated state survives the first new method.

The most common failure is testing the software instead of the intended use: three hundred screenshots proving that a button saves a record, and nothing demonstrating that an out-of-specification result cannot be released without a second signature. The second is treating validation as a launch activity, so the evidence describes a system nobody has run since. The third is the interface nobody scoped, where a validated instrument writes into a validated LIMS through a step a person performs by hand.

All three are cheaper to prevent than to remediate, and all three are visible in a scoping conversation before anyone writes a protocol. Laboratory change control is the service that addresses the second and keeps the third from reappearing.

What can a vendor do, and what stays yours?

The honest boundary matters here, because a vendor claiming to validate your system for you is claiming something it cannot deliver.

THE VENDOR CAN SUPPLY	ONLY THE LAB CAN SUPPLY
Platform and installation qualification evidence	The intended use statement
Qualification protocols and templates	The decision about which risks deserve testing
Evidence of its own development testing	Performance qualification against your workflows
People who have done it before, and the configuration work	The approval signatures
A change-control process and the tooling for it	The judgement, each time, of what a change puts at risk

If you are scoping this now, the useful first question is not which vendor validates faster. It is what your intended use statement says, because every hour of testing after that is either aimed at it or wasted on something else. Validation services describes how the left-hand column is delivered; compliance services covers the documentation set the whole exercise produces.

What a LIMS carries in this work

The evidence, and the controls the evidence is about. A LIMS keeps the audit trail that shows the prior value, attributes every entry to a person and a time, enforces the second signature before a result leaves, and records each configuration change so that the validated state can be shown to have been maintained rather than merely declared. What it cannot carry is the intended use statement, because that is a claim about your laboratory that only your laboratory can make.

> **How LabLynx handles this**
>
> LabLynx runs validation as a service engagement alongside implementation: protocols and templates, qualification support, and the configuration work that produces the evidence in the first place. It also runs the compliance and change-control side, which is what keeps a validated state valid after the first method changes. What stays with your lab is the intended use statement, the approval signatures, and the decision about which risks deserve testing. That division is not a limitation of the service; it is what the regulations require, and we would rather say so than sell you a certificate that answers none of an inspector's three questions.

Frequently asked questions

Can a LIMS be sold as pre-validated?

No. A vendor can test its own software and hand you that evidence, and it can supply protocols and templates that shorten your work considerably. What it cannot do is validate your configuration, your workflows and your intended use, because those did not exist until you built them. A claim of "pre-validated" describes the vendor's development testing, which is useful and is not your validation.

How long does LIMS validation take?

It scales with the risk and the number of configured workflows, not with the size of the software. A lab with a clear intended use statement and a handful of high-risk workflows can complete qualification in weeks alongside configuration; a lab that starts testing before writing the intended use statement will take months and produce evidence aimed at nothing. The statement is the schedule.

What is the difference between validation and qualification?

Qualification is the evidence for one layer: that the software was installed as specified, that a function operates as specified, that a workflow performs as your lab requires. Validation is the conclusion drawn across all of it that the system is fit for its documented intended use. A vendor can supply installation

qualification and much of the operational layer; performance qualification is yours.

Does adding a new test method mean revalidating everything?

No, and that is what change control is for. A change is assessed for risk, and testing is aimed at what the change could affect: the new method, anything sharing configuration with it, and the interfaces it touches. A well-kept validated state makes that quick because the intended use statement already says what matters. Without change control, every change is a small undocumented revalidation.

Is 21 CFR Part 11 the only rule that applies?

Only if your records support FDA-regulated products and nothing else. An accredited testing laboratory answers to ISO/IEC 17025, which requires information management systems validated before use and after change. An EU GMP site answers to Annex 11. Device production software falls under FDA's computer software assurance guidance. Most projects are actually run to GAMP 5, which is guidance rather than regulation.

What does an inspector ask about a computerised system?

Three things, in roughly this order: what the system is for, who approved that statement, and what testing supports it. Then they follow the audit trail for a specific record and ask who could have changed it and whether the prior value survives. A vendor's certificate answers none of the first three, which is why the intended use statement, signed by the lab, decides the conversation.

Sources and references

1. 21 CFR Part 11, Electronic Records; Electronic Signatures

   Electronic Code of Federal Regulations

2. EudraLex Volume 4, Good Manufacturing Practice guidelines, Annex 11: Computerised Systems

   European Commission

3. Computer Software Assurance for Production and Quality System Software, Guidance for Industry

U.S. Food and Drug Administration

Brandon Holland

Marketing & Operations Digital Solutions Architect

Brandon Holland is LabLynx's Marketing & Operations Digital Solutions Architect. He builds and runs the digital systems behind the company's public surface, from the website and content architecture to the tooling that keeps product information accurate everywhere it appears. He writes about laboratory informatics from the systems side: how lab records get structured, searched, and kept defensible as a lab scales.

Read the current version online

https://www.lablynx.com/resources/articles/lims-validation-what-regulators-require/

Downloaded September 8, 2026. LabLynx builds laboratory informatics software: LIMS, ELN, LIS and lab automation.