

TURNAROUND STARTS IN THE INBOX

IntelliParse for labs: an ingestion and workflow engine built around the document flow you already have

For lab directors, operations managers, and LIMS owners at testing, clinical, and analytical laboratories

The sample is in the building. The order is not. It's in an inbox, a fax tray, a client portal nobody has logged into since lunch, or a PDF attached to an email that says "see attached." Accessioning can't start until someone types what that document already says — and the turnaround clock your client is watching started at drop-off, not at data entry.

Every lab knows this queue. Nobody budgets for it, because it isn't a line item. It's the gap between when work arrives and when your systems know it arrived. IntelliParse closes that gap without replacing anything you run today.

WHAT A CAPABILITY ENGAGEMENT IS — AND ISN'T

IntelliParse is **not** another platform to log into, and it is **not** a new LIMS. It's a capability we build *around* your existing document flow: an ingestion and workflow engine that reads what comes in, checks it, and hands clean data to the systems you already have.

Keep the LIMS. Keep the portal. Keep the way clients send you work. The engine sits in front of them and does the typing, the checking, and the routing — with your people reviewing the exceptions instead of every page.

You own the source code and data we build for you. Slingr hosts and fully manages it on our cloud, in a dedicated project for your lab — data stays isolated, and there's no training on your data. It's code your team can read and extend, not a vendor release cycle you wait on. And AI is metered: you pay for what the pipeline actually uses, not per seat or per page.

HOW THE PIPE WORKS IN A LAB

1. Documents arrive — in whatever form they already arrive. Requisitions by email, orders from a client portal, chain-of-custody forms as scans or phone photos, subcontract results from a reference lab, supplier certificates, spreadsheets. One intake, every format. Your clients don't have to change a thing.

2. The engine reads them into your schema. Client, sample IDs, matrix, panel or method codes, due dates, preservation and hold times, signatures on the COC — extracted into the fields your LIMS expects, structured the way your lab names things.

3. Deterministic checks run first; a person sees only the exceptions. Does the client exist? Is that panel one you run? Is the COC signed and the relinquish time plausible? Rules catch the obvious before a model is ever called, and anything that still looks wrong lands in a review queue for a human — not silently in your records.

4. Clean data is routed where it belongs. An accessioning record is created in the LIMS with the source document attached, the client gets confirmation, and the sample is ready for login the moment it hits the bench. The same pipe runs outbound: results and COAs into the client portal or ERP, invoices matched to POs.

WHERE LABS FEEL IT FIRST

DOCUMENT	TODAY	WITH THE PIPE
Requisitions and orders	Re-keyed into the LIMS after someone opens the email	Accessioning record created on arrival; exceptions queued
Chain-of-custody forms	Scanned, filed, cross-checked by hand	Fields captured and attached to the sample record with the original image
Subcontract and reference-lab results	Transcribed from PDF into the report	Parsed to your result schema and flagged if units or limits don't match
Supplier certificates and standards	Tracked in a spreadsheet, found at audit time	Lot, expiry, and traceability captured and watched for expiration
Client POs and invoices	Matched in accounting at month end	Matched as they arrive; only the mismatches reach a person

FIVE THINGS TO INSIST ON — FROM US OR ANYONE ELSE

- 1. Exceptions go to a person, not into the record.** If a field doesn't validate, a human decides. Confidence scores are not approvals.
- 2. Rules before models.** Don't pay AI tokens to check that a client ID exists. Cheap, deterministic checks run first; expensive models are reserved for the pages and fields where they actually move accuracy. This is also how the AI bill stays proportional to your volume.
- 3. Your schema and your LIMS — not theirs.** The engine writes what your systems expect. No templates you have to force documents into, no new system of record.
- 4. Code you can read and extend.** A new client format, a new panel, a new destination — your team or ours extends the pipeline directly. You are not filing a feature request.
- 5. The audit trail is yours.** Every extracted field traces back to the source document and to whoever approved the exception. It lives in software you own, retained on your schedule.

WHAT THIS LOOKS LIKE IN PRACTICE

For a diagnostic lab client, Slingr delivered the system in **three months with a single developer**, and manual data entry was **reduced to near zero**, with faster result turnaround and improved billing accuracy. For a genomics testing lab, sample management and report generation were automated and clients got a self-service portal. Different labs, same pattern: the documents already say what they say — the software should be the one reading them.

Slingr builds IntelliParse as a capability engagement: an ingestion and workflow engine around the document flow your lab already has, feeding the LIMS you already run — hosted and fully managed by Slingr in a dedicated cloud project, in code you own outright. Tell us how your orders come in and we'll draft a real plan, same day: slingr.io/labs-estimate