

For Luís — what the .NET side needs decided, and what it gives back

From: Miguel (GOTT.Sibyla) · Date: 2026-08-03 · Read time: ~10 minutes

Why you're getting this

On 2026-08-03 the Sibyla .NET backlog was reissued as **Development Plan v5.0** (docs/project-todo.md in the GOTT.Sibyla repository). It merges the previous v4.0 plan with your Stage 8–10 work. Three things changed as a direct result of what you built:

- FDR Layer 5 moved out of Phase 2 into Phase 1.** Bank movements, reconciliation and RECREV are now in the first release rather than parked. Keeping them parked would have forked the queue model and the flag machinery between the document side and the bank side.
- DOCRQE is specified as a persisted queue**, not the recomputed live view v4.0 assumed. Your Stage 9 R7 conversion settled that argument.
- 22 of the 133 backlog items are marked [S]** — meaning a working reference implementation exists in your prototype and the .NET job is to *port proven behaviour, not redesign it*. P1-5 (flag governance), P1-6 (queues), P1-7 (apply engine) and P1-11 (bank reconciliation) are mostly yours already.

That last point is the reason for this note. Your prototype has stopped being a prototype — it is now the specification for a chunk of the production system, and a few things need to be pinned down before the .NET schema is frozen.

A correction owed to you up front. The v5.0 plan was published this morning with "134 stale DOCLOG rows sharing 35 LGCodes" written up as a hard blocker on the .NET migration. Your 10:05 correction had already taken that finding apart more accurately than we had, and we were still working from the pre-correction text. The plan has since been amended. Nothing below assumes the old framing.

Where things actually stand (your 19:30 numbers)

For reference, so we are working from the same figures. The v5.0 document was written against your 09:30 state and is roughly a day stale in places:

	V5.0 AS PUBLISHED (09:30)	YOUR STATE AT 19:30
FDCHDR	1,096	1,158 (I 481 · R 245 · P 46 · F 77 · O 309)
2026 reconciliation	91.8%	94.6%
Actively-blocking flags	12	0
DOCRQE open	2,627	164 real decisions of 180 Decision-class
RECREV open	270	153

The plan has since been refreshed to these figures (`project-todo.md` rev. 2, `project-evolution.html` , and the v13 flow diagram all updated 2026-08-03 evening). Tell us if any are still moving.

A. The identity problem — you closed it, and here is what we took from it

When this note was drafted, the three code-identity defects were the only thing gating the .NET schema freeze. **You closed all three overnight** (Roadmap 35, 36 and 29), so nothing is being asked of you here — this section records what the .NET side is porting, so you can correct us if we have read any of it wrong.

- **P/F/O documents key on the movement's natural key.** We are porting `SourceKey` and the exclusion of `SourceFile` from `BMCode` , and preserving the two details your tests surfaced: the comma-joined `BMCode` lists on the seven aggregated pay-run documents, and the recomputed occurrence tiebreaker for the byte-identical same-day BCP lines.
- **DOCLOG keys on (Filename, EntryCode)** . `LGCode` is the per-row identity, so the separate `DocLogId` we floated is dropped.
- **PAYCODE / RCVCODE from the fiscal document plus payment sequence.** We are also porting `verify_code_identity_stability.py` itself, including the row shuffle — the point that an order-preserving test goes green while the defect is live is the part worth keeping.

B. Decisions taken on the .NET side — flag anything you disagree with

B.1 Is `ItemClass` final?

Your 18:10 round added `ItemClass` (Decision / Status / Annotation), derived at creation from what a human could actually change, recomputed every run. It is going straight into the .NET queue schema as a first-class column, and the review UI will be built to show only Decision-class items as a backlog.

Decided: frozen at three. We are modelling it as a closed enumeration rather than an extensible lookup, so every .NET detector must declare which of the three it emits and must fit one of them. A fourth class later is a migration — and we are treating that friction as a feature, since it is what stops the phantom backlog re-forming. Tell us if you think a fourth is coming; otherwise nothing needed.

Related — your five "Preventing phantom backlog" rules are being adopted as .NET Definition-of-Done items, in particular **Rule E** (a detector must state which class it emits before it is added). Flag it if you'd rather they stayed FDR-side conventions.

B.2 "Reconciled is not recorded"

Your standing control — a movement matched to a *ledger* reference rather than an `FDCHDR` `EntryCode` counts as reconciled in every metric while leaving no financial entry behind it — is the kind of thing that should be a hard validation rule in the .NET pipeline, not a convention. It is what hid the 2026 Jan–Jun payroll gap (€142,835.50, six months, and nothing flagged it because the movements looked matched).

Decided: it gets its own BNKMAT match status (`Matched – ledger reference, no entry`), excluded from the reconciliation percentage. We took the structural route over a flag on an otherwise- `Matched` row, because a separate status makes the movement impossible to count as clean by construction rather than relying on whoever reads the metric to filter. A `DOCEFL` rule can be layered on later without a schema change if the status alone proves too quiet.

B.3 History import — opening balances or fresh start?

P1-0 has to decide whether the .NET system imports your history (DOCLOG ~1,466, FDCHDR 1,158, FDCDTL) as opening balances, or starts fresh with reference data only (ENTMST, ENTALS, ITMMST, ITMALS, ENTITM, ITMCLS, MNGACC, SNCACC, COCACC, EXCRAT, DOCTYP and DOCEFL rules).

Decided: full 2026 history as opening balances, plus the complete reference layer. 2025 stays out, following your scope rule. We accept the consequence that the Toorist related-party position cannot close across years until a later full-history round.

One thing we would still like your read on: **is the 2026 Jan–Jun payroll rebuild stable enough to import?** It is six months of entries that did not exist two days ago, and it comes in as opening balances if we go now.

B.4 The company dimension (your Roadmap 33)

You found the shared inbox has no company signal, and that capturing by *location* rather than *content* would have booked Itoorer revenue as a Gott cost. Your proposed fix — a capture-time assertion that the document's own Bill-to/issuer VAT number matches the company it is booked to — is already in the v5.0 plan on the .NET side as the `counterparty_not_internal_company` gate.

Question: do you want it built FDR-side too, or is the .NET gate sufficient? If the prototype keeps ingesting into 2026, it probably needs its own copy.

C. Items that are yours alone, listed so nothing is lost

These need a decision from you or from the business, and nothing on the .NET side depends on them:

- **Hydra iT — Fatura FV2502560.PDF** (2025-11-13), genuinely uncaptured, outside 2026 scope, explains no 2026 cash line. Capture or formally exclude. (*Roadmap 34*)
- **Doc_GBA_FA_48_2026_Signed.pdf** — parked as redundant, but a byte comparison shows it differs from the archived copy at the same size, almost certainly the signed variant. Restore from `_to_delete` if the signed copy should be the retained one.
- **Diego Berlitz** — the one remaining duplicate-identity CONFLICT.
- **SNCACC against LMD's chart of accounts**, and the 6 ITMMST rows with blank SNCACC — needs the accountant. (*Roadmap 26*)
- **202601–202606 cross-check** against the accounting records. (*Roadmap 27*)
- **Toorist / Itoorer intercompany stream** — the accounting treatment (shareholder loan, current account, or capital contribution) is what actually resolves those lines; the ignore-for-matching rule is a holding position.
- **Missing reference data:** 75 provider entities without `InvoiceFrequency`, COCACC still one placeholder row, 27 FDCHDR rows at `FEX = NO RATE`.

D. What the .NET side gives back

Not everything flows one way.

- **Permanent identity, solved structurally.** The three defects in section A stop being possible once codes are issued from database-backed sequences that never reassign. You will not have to keep catching renumbering after the fact.
 - **EnteredBy / EnteredAt everywhere.** Your KL-38 — no actor identity anywhere in the registry, which the policies require but a single-operator pipeline cannot honestly supply. The .NET harness has authenticated users and append-only audit natively, so this closes on migration rather than needing prototype work.
 - **The Discard / Purge lifecycle.** Being designed in P1-0: reviewer *Discard* is soft (excluded everywhere, bytes and audit retained); authorized-role *Purge* deletes bytes and leaves a tombstone with metadata and SHA-256; never automatic, never available for anything that reached a fiscal or archive table. It is being written so it is usable from your side too — it is the mechanism your never-delete rule has been missing. Your 10:05 correction is the argument for it: a blanket delete would have destroyed the only surviving trace of the payroll gap, and a soft-discard-with-evidence would not have.
 - **rm and the archive.** The .NET archive writes the business copy and confirms it before the document transitions to Archived, with an exactly-one-copy proof under crash and retry. The `_to_delete` workaround goes away.
-

E. One ask about pace

Now that 22 backlog items are being ported from your code rather than specified independently, **structural changes to the prototype ripple into the .NET design**. Not a request to slow down — the Stage 10 work today is exactly the kind of thing we want found before it is cast in a migration.

What would help: a heads-up when you change **a table's shape, a code's meaning, a decision vocabulary, or a sheet's identity** — the DOCTYPE → DOCTYP rename and the new `ItemClass` column are both good examples. Behaviour fixes, new detectors, tolerance tuning and data corrections need no notice at all.

The short version

#	ITEM	STATUS
A1	Natural key for bank-generated P/F/O	Closed by you — SourceKey , SourceFile excluded
A2	LGCode vs a separate DocLogId	Closed by you — LGCode re-keyed on (Filename, EntryCode); no DocLogId
A3	RCVCODE stability	Closed by you — 0 of 245 move under shuffled cold rebuild
D4	History import scope	Decided — full 2026 + reference layer, 2025 out
D5	Reference Only document state	Decided — first-class state in the .NET schema
D6	"Reconciled is not recorded"	Decided — own match status, excluded from the rate
D7	Grandfathered blocking flags	Decided — import with DetectedAt , enforcement stays prospective
D8	Is ItemClass final?	Decided — frozen at three; a fourth is a migration
D9	Recurring payment-control imports	Decided — additive, keyed by period; re-import replaces only that period
B4	Company-dimension gate FDR-side too?	Your call
—	Is the 2026 Jan–Jun payroll rebuild stable enough to import?	Your call
C	Signed GBA variant, Diego Berlitz, SNCACC, 202601–06 cross-check, intercompany treatment, missing reference data	Yours / accountant / business

Nothing on the .NET side is waiting on a decision. The two questions marked *your call* shape work rather than block it.