ALLIED BizTech SOLUTIONS

FIELD GUIDE · 2026

Rebuild Notion yourself.

The plan we follow.

For teams of 200 to 2,500+ seats who want to own their wiki, editor and search on their own AWS, keep every page they have written, and stop paying per seat.

- → Is it worth it, at your size
- → What to rebuild and what to drop
- → The architecture, in three layers
- → Migration and go-live checklists
- → The data model
- → A week-by-week build plan
- → Permissions, search, the editor
- → Where teams get stuck

Notion charges per seat. *A wiki costs bytes.*

Notion's per-seat model makes every new hire a budget line, and the seat price steps up from Plus to Business to Enterprise. Self-hosted Outline or Affine costs scale with storage and traffic, both far cheaper. The bigger your team, the sharper the math.

SEATS	NOTION A YEAR	REBUILT: YEAR ONE / AFTER	SAVED IN YEAR ONE
50 seats · Plus	~$6,000	~$16,700 / $1,800	Stay on Notion
200 seats · Business	~$60,000	~$17,300 / $2,400	~$42,700
500 seats · Business	~$150,000	~$20,900 / $6,000	~$129,100
1,000 seats · Business	~$300,000	~$26,900 / $12,000	~$273,100
2,500 seats · Enterprise	~$900,000+	~$65,900 / $36,000	~$834,100

Notion list prices, May 2026. "Rebuilt" runs Outline (BSL, free for internal use) on AWS EC2 with Postgres and S3, plus our fixed deploy price: $14,900 up to 1,000 seats, $29,900 (the top of $19,900–$29,900) from 1,000 to 5,000. If your own team builds it, put your engineering cost in that place.

THE RULE OF THUMB

At 50 seats, stay on Notion: the deploy does not pay back. From 200 seats it pays back in months. **At 500 seats, the rebuild saves ~$129K in year one and ~$144K a year after.** From 1,000 seats, year one alone saves more than $250K.

Who you need

- **A senior infrastructure engineer** at home with Docker, Postgres, S3, Redis and your identity provider. Production-grade ops, SSO, SCIM and content migration take 3–4 weeks of that person's time.
- **Your workspace owner or IT lead**, two hours a week: the spaces that matter, the templates, the parallel-run sign-off.
- **An owner after launch**: upgrades, backups, permissions, search quality and on-call.

02 · SCOPE

Rebuild the 15–25% you use. *Drop the rest.*

Notion is a block-based wiki plus a database tool. Most teams live in the wiki: pages, the editor, search and templates, and Outline matches that closely. Databases are where you decide, not copy.

REBUILD ONE FOR ONE

- **Workspaces and collections**: unlimited nesting, breadcrumbs
- **Block editor** with the same slash commands
- **Real-time collaboration** on Y.js: presence, cursors
- **SSO and SCIM** from Okta, Azure AD or Google
- **Permissions**: private, team, public; workspace and page level
- **Search** across every doc, permission-aware
- **Templates**: your 5–15, from meeting notes to OKRs
- **An API shim** for your 2–5 internal tools on Notion's API
- **Import** of 95%+ of typical Notion content

REBUILD, SIMPLER

- **Databases**: tables, board and timeline views; complex ones on Affine, Linear or Airtable
- **Notion AI**: Anthropic or OpenAI direct, on selected docs
- **Embeds**: Figma, Loom, YouTube work; a few re-created

DROP ON PURPOSE

- **Notion Sites**: Webflow, Framer or Astro
- **Notion Calendar**: Cron, Calendly, Google Calendar
- **Formulas, relations, rollups**: Airtable
- **Native mobile editing**: web-responsive by default
- **Guest sharing**: Box, Dropbox or your CRM portal

ADD WHAT PER-SEAT PRICING DISCOURAGES

- Unlimited users; the bill follows bytes
- Air-gap or on-prem for regulated teams
- Direct SQL from your warehouse

Three layers. *Same wiki UX.*

Outline is the closest match to Notion at the wiki layer; Affine adds whiteboards and databases. Pick by your team's real usage, not the feature list, and run either on the same infrastructure in your AWS.

OPTION A · WIKI-HEAVY

Outline (BSL)

- Free for internal use
- Workspaces and collections
- Rich-text blocks, slash commands
- Real-time collaboration (Y.js)
- SSO via OIDC or SAML
- A Notion importer

OPTION B · DATABASES

Affine (MIT)

- Fully open source
- Block editor, whiteboard, databases
- Closer match to Notion databases
- Newer, with less mature ops
- Often blended with Outline

SHARED · EVERY DEPLOY

Your infrastructure

- **Postgres**: content, users, permissions
- **S3**: uploads and attachments
- **Redis**: presence and locks
- **Your IdP**: SSO and SCIM
- **Meilisearch**: the search index
- An audit log in your database

Decisions to make in week one

DECISION	OUR DEFAULT	WHY
Which editor	Outline for the wiki; Affine for databases	Decided from a usage audit
Where it runs	Your AWS account; on-prem if regulated	Content and access logs stay yours
Sign-in	SSO and SCIM from your IdP	Leavers lose access the day they leave
Search	Meilisearch, filtered by permission	Fast full-text with no seat fee
Files	S3 behind signed, short-lived URLs	Attachments follow page permissions
Backups	Daily snapshots of Postgres and S3	A restore you have proven, not assumed

Licenses: Outline is BSL 1.1, free to self-host for internal company use; Affine is MIT, fully unrestricted. Running either for your own company on your own AWS carries no seat fee.

Nine tables *carry the whole wiki.*

Outline and Affine bring their own schema; this is what it holds and what your import has to fill. Keep Notion's ID on every imported row, so the import can run twice and any page traces back to its source.

TABLE	KEY COLUMNS
`collections`	id, notion_id, name, parent_id, default_permission, sort, created_at
`documents`	id, notion_id, collection_id, parent_document_id, title, template_id, created_by, updated_by, updated_at, archived_at
`document_state`	document_id, ydoc_state (the Y.js document), version, updated_at
`revisions`	id, document_id, body (Markdown), author_id, created_at
`users`, `groups`	id, idp_subject, scim_external_id, email, name, role, suspended_at
`memberships`	id, principal (user or group), object (collection or document), permission (read, comment, edit, manage), inherited, source (notion, scim, manual)
`attachments`	id, document_id, s3_key, content_type, size, notion_url
`redirects`	notion_url, notion_id, document_id: every old link, resolved
`audit_events`	id, actor_id, action (view, edit, share, export, delete), document_id, ip, at

TWO RULES THAT SAVE A REBUILD

Rewrite every internal link from Notion's IDs: a migrated wiki with broken links stops being read. **Check permissions in one place**: the page, search results, file downloads, the API shim and AI answers all call the same check, so no route shows what the page would hide.

Five weeks, *parallel run included.*

This is the order we build in. Each week ends with something your team can open, and the last week runs both workspaces side by side before Notion goes quiet.

WEEK	BUILD	DONE WHEN
1	**Audit and foundations.** Which spaces, databases, templates and integrations are in real use; the Outline-or-Affine call. AWS, Postgres, S3 and Redis in Terraform; SSO and SCIM.	Pilot users sign in through your IdP
2	**Export and import.** Every page and database through Notion's API as Markdown and JSON into staging: hierarchy, permissions, attachments to S3, internal links rewritten.	Page counts match; 50 pages spot-checked
3	**Search, templates, integrations.** The permission-filtered index, the 5–15 templates, the API shim for internal tools, AI on selected docs.	Search returns only pages the user can open
4	**Databases and the last 5%.** Basic tables and views; complex databases to Affine, Linear or Airtable; synced databases and niche embeds by hand. Backups, DR, a load test.	A restore from snapshot is proven
5	**Parallel run and cutover.** Both workspaces live for a week with a delta import and a daily fix list; then old links redirect.	The team works a full week without opening Notion

AFTER CUTOVER

Keep Notion **read-only until your renewal date**, then cancel. Teams adapt in 1–2 weeks, because the slash commands they know (`/h1`, `/code`, `/quote`) work the same.

Who sees what, what search finds, *how typing feels.*

Installing a wiki is a day's work. People trust it, or drift back to shared drives, on three things: a private page stays private, search finds the runbook, and two people can type on one page without a stutter.

1. **Map Notion's permissions, then simplify.** Notion mixes workspace, teamspace and page-level shares with guests. Map each to collection or document memberships, list every page shared with a guest (guest sharing moves to a separate share layer), and check inheritance on a sample before the full import.
2. **One permission check, everywhere.** The page, search, file downloads, the API shim and AI answers all call it. Filter search inside the index by the reader's groups, never after results come back, and re-index a page the moment its permissions change.
3. **Search people trust.** Index titles, headings and body on every save; weight titles and recent edits. Review zero-result searches weekly: each one is a missing page or a missing synonym.
4. **An editor that feels like home.** Outline and Affine are block editors on Y.js. Keep the slash commands people know, test paste from Google Docs, Word and Slack, and load-test many editors on one page. Store the Y.js state for live editing and Markdown revisions for readable history.
5. **Links and embeds that survive.** Resolve every notion.so link through the redirects table, and re-create the niche embeds by hand before the parallel run, not after.

Notion AI, scoped

Anthropic or OpenAI on your own account, over the docs you select, behind the same permission check. You decide which content the model sees, and you pay tokens, not seats.

Phones

The wiki ships web-responsive. In the parallel-run week, check the pages people read on phones, such as runbooks and on-call notes.

Move it, launch it, *run it.*

Migration

- ☐ Audit usage: spaces, databases, templates, integrations
- ☐ Export every page and database through Notion's API
- ☐ Keep Notion's ID on every page and file
- ☐ Create collections, groups and memberships before content
- ☐ Copy attachments to S3 and rewrite their links
- ☐ Rewrite internal links; fill the redirects table
- ☐ Re-create synced databases and niche embeds by hand
- ☐ Check counts, then read 50 pages end to end

Go-live

- ☐ SSO and SCIM live; leavers lose access the same day
- ☐ Search tested with a restricted user for leaks
- ☐ Daily snapshots and a proven restore
- ☐ Internal tools pointed at the API shim
- ☐ Rollback ready: Notion read-only, one switch away

Running it

- ☐ **Always on:** uptime alerts, security patches
- ☐ **Weekly:** zero-result searches and failed imports
- ☐ **Monthly:** Outline or Affine upgrades through staging
- ☐ **Quarterly:** access review, cost review against the Notion bill you replaced
- ☐ **Yearly:** a restore drill, a license and permission audit

Who owns it

Name one person before launch. The wiki is the company's memory: someone has to answer when search stops finding the runbook at 2 a.m.

Five ways a rebuild *stalls.*

1. Switching brands, keeping the seat fee

Confluence, Coda, ClickUp, Slite and Slab: same per-seat pricing, different brand. Confluence Cloud is comparable to Notion Business per seat, and Atlassian Data Center self-host runs $30K+ a year and is clunky to operate.

2. Coda for the databases

Coda's databases are stronger than Notion's, and pricing per doc maker is interesting at small scale. At enterprise size the math converges with Notion's.

3. The $8K freelancer

You get docker-compose up. SSO, SCIM, search, content migration, backup and DR are still on you, and they are 80% of the value.

4. Self-hosting Outline and calling it done

It is open source and you can. Production-grade ops, SSO, SCIM and content migration are typically 3–4 weeks of senior infra time. Plan for them.

5. Copying every database into the wiki

Complex formulas, relations and rollups belong in Airtable, Linear or Affine. Decide each database's home in the audit, and the wiki ships on time.

And one that is not a mistake: negotiating

Notion discounts Business and Enterprise 15–25% on multi-year, multi-thousand-seat commits. Do it if the rebuild is a year away; it captures less than a quarter of what self-hosting does.

ALLIED BizTech SOLUTIONS

RATHER HAVE IT BUILT?

We build this exact plan. *Fixed price.*

The same scope, stack and five weeks, delivered by a senior team that has shipped 250+ products.

$14,900 fixed up to 1,000 seats, $19,900–$29,900 for 1,000–5,000, $34,900–$59,800 above 5,000; a databases module for $6,600–$11,600

Live in 4–5 weeks, with a one-week parallel run next to Notion before cutover

Every page moved through Notion's API, with hierarchy and permissions; the last ~5% re-created by hand

90-day warranty, SSO, SCIM, search, daily snapshots and a runbook, all on your AWS

We can run it for $3,600 a year: monitoring, monthly upgrades and SLA support

alliedbiz.tech/rebuild/notion-enterprise-alternative

Allied BizTech · 15+ years · ISO 9001:2015 · 250+ products shipped. Notion is a trademark of its owner, named here for identification only; this guide is independent and not endorsed by Notion. Prices as of May 2026.