

A JOB READINESS PLAN FOR SAP ECC END OF MAINSTREAM MAINTENANCE 2027



If your batch schedule runs on SAP ECC, the platform underneath it is approaching a critical maintenance expiry date. Most guidance on the 2027 deadline treats the ERP migration as a whole — the data model, the custom code, the functional scope. This guide is narrower and more operational: it's the plan for the team that owns the jobs. It walks through five steps to take your scheduling estate from an undocumented ECC dependency to a mapped, rationalized, portable workload that moves with you — whichever target you choose.

What the 2027 deadline changes

SAP has set firm dates for the end of mainstream maintenance on SAP ERP 6.0, the core of SAP Business Suite 7. For the enhancement packages most customers run today (EHP 6 through 8), mainstream maintenance ends on December 31, 2027, followed by optional extended maintenance through the end of 2030 at a premium on the maintenance base. The older EHP 0–5 versions already reached the end of mainstream maintenance on December 31, 2025, so those systems have moved into customer-specific maintenance — a phase whose reduced scope no longer includes legal and regulatory updates. Eligible large enterprises moving especially complex programs to cloud infrastructure have one further option: a private-edition transition via RISE that extends ECC operation for select customers, subject to SAP's eligibility, architecture, commercial, and transition requirements.

Whatever path you choose, the scheduling estate needs an explicit plan: [migrate to a supported target](#), redesign it for a new interface, or retire it. That work runs in parallel with the functional project

— it has its own critical path, and the five steps below are that path.

Step 1 — Inventory: map every job before you move any

You cannot migrate what you haven't mapped, and many ECC estates can't produce a complete map on demand. Jobs scheduled directly by users in SM36, the SAP transaction for defining background jobs, remain visible in SAP but sit outside any central catalog, so the true scope of a migration is almost always larger than the documented one.

The first move is to bring every unmanaged job under [central governance](#). Job extraction and mirroring pull user-scheduled SAP jobs into a single managed catalog, so the scope you plan against is real rather than assumed. This is the foundation for surfacing hidden dependencies before they become cutover issues.

Step 2 — Rationalize: shrink the estate before you carry it

A smaller system migrates faster. Years of accumulated housekeeping jobs, one-off reports, and superseded processes increase scheduling complexity. Unnecessary application data also increases the volume processed during archiving, system-copy, backup, and migration activities — delaying the S/4HANA roadmap by lengthening every technical step along the way. Reducing unnecessary data and obsolete scheduling logic can simplify the migration scope and reduce effort. Two actions slim the estate:

- **Archive with SARA.** Data archiving runs — write, delete, and store across archiving objects — shrink the underlying tables that the migration must process.
- **Retire the housekeeping jobs running on habit.** The central catalog from Step 1 finally lets you see the full set of recurring technical and administrative workloads across the estate. That visibility makes it easier to identify legacy schedules, duplicate processes, obsolete reports, monitoring tasks, and maintenance jobs that no longer provide business value. Review each workload deliberately and retire what is no longer needed before carrying it into the target environment. Our guide to background job scheduling at enterprise scale walks through the housekeeping workloads that typically accumulate and how teams bring them under governance.

Step 3 — Choose the target topology

"Migrating off ECC" is not one destination. The scheduling implications differ across four common landing zones, and the interface your jobs run through changes with each. Decide the target before you convert — it determines everything downstream. Most of these targets expose XBP, SAP's certified external interface for background processing; one does not.

Target topology	What it is	What changes for your jobs
On-premises S/4HANA	S/4HANA in your own data center	Same certified XBP interface; job definitions carry over unchanged
RISE with SAP S/4HANA Cloud Private Edition	S/4HANA in an SAP-managed private cloud	XBP stays exposed over a secured remote connection; a minor config change

Private edition plus SAP BTP	S/4HANA with extensions on SAP BTP	XBP core jobs plus BTP Job Scheduler jobs, coordinated together
GROW with SAP S/4HANA Cloud Public Edition	S/4HANA Cloud, public edition	No XBP exposed; scheduling shifts to the BTP Scheduler and the External Scheduler API (SAP_COM_0948)

The through-line: the same SAP job type spans SAP R/3 and S/4HANA, so an S/4HANA move needs no new plugin and a RISE move is a minor configuration change. The one meaningful shift is GROW public cloud, where the scheduling surface is different. Our RISE, clean core, and GROW guide compares the four topologies in full, including agent placement, encryption, and the high-availability model for each.

Step 4 — Convert instead of rebuild

The instinct on a platform change is to rebuild the schedule by hand in the new environment. On an estate of any size, that's slow, error-prone, and throws away the dependency logic you spent years getting right — and it isn't necessary.

A conversion tool imports existing job definitions directly — jobs and calendars together — and produces an assessment report enumerating what it found, following a defined sequence: select the project, evaluate the data, run the conversion, validate the results, and load into the target. Converting rather than rebuilding preserves the cross-system logic — the dependencies that tie an SAP job to the file transfer, the bank, or the data platform waiting on it — instead of asking a person to reconstruct it from memory. Our SAP batch jobs and ECC-to-S/4HANA migration guide covers the conversion mechanics in full.

Step 5 — Simulate and protect the cutover

The last risk is cutover weekend itself, and it has two failure modes: a batch run that behaves differently on the new platform, and a job that writes into SAP while Basis work is mid-flight.

- **Simulate the run.** Forecast and What-If dry-run the batch schedule against the new environment before go-live, so you see the shape of the run — and where it breaks — before it's carrying real data.
- **Protect the window.** A global stop, hold, and restart across [the entire job estate](#) ensures nothing posts into SAP during the upgrade, support-pack, and enhancement work a conversion weekend involves, then releases the estate cleanly once the window closes. Upgrade and support-pack projects are a well-known source of business-process disruption precisely because jobs keep firing into a system that isn't ready for them; a global hold materially reduces the risk of that happening.

Build your timeline

The deadline is fixed, so plan from it. S/4HANA program duration varies widely by scope, landscape complexity, data strategy, custom code, deployment model, and rollout approach, so work backwards from the deadline and your organization's validated program plan.

The arithmetic is tighter than it looks. As of August 2026, roughly seventeen months remain before mainstream maintenance ends for EHP 6–8 systems. Many S/4HANA programs run twelve to twenty-four months depending on scope, which means the shorter end of that range still fits inside the window and the longer end no longer does. A program that hasn't started scoping is, in practice, already planning around extended maintenance as a bridge rather than around the 2027 date itself — and that bridge carries a premium on the maintenance base.

For the scheduling team, the sequence is the same regardless of target:

- **Inventory** — map and centralize every job
- **Rationalize** — archive and retire
- **Choose topology** — on-prem, RISE private, RISE + BTP, or GROW
- **Convert and parallel-run** — import, validate, [run old and new side by side](#)
- **Cutover** — simulate, protect the window, release

Steps one and two return value immediately — a mapped, slimmer estate is easier to run today, deadline or not — which is the argument for starting them now rather than waiting for the migration project to formally begin.

Frequently asked questions

What happens to my SAP batch jobs when ECC support ends in 2027?

The maintenance milestone does not automatically stop your SAP system or its batch jobs — they keep running as long as the system runs. What ends is SAP's mainstream maintenance of the platform beneath them, on December 31, 2027 for EHP 6–8 systems; the older EHP 0–5 deadline passed at the end of 2025. The practical work is not to the jobs but to the estate: [inventory every scheduled job](#), including the ones users created locally in SM36; archive and retire what no longer earns its place; then convert the definitions onto your chosen S/4HANA target rather than rebuilding them by hand, so the cross-system dependencies survive the move. That work runs in parallel with the functional S/4HANA project and is best started well ahead of the deadline.

Do I have to redefine every SAP job when I move to S/4HANA?

No — with one exception. The same SAP job type spans SAP R/3 and S/4HANA, so an S/4HANA move needs no new plugin, and a move to RISE with SAP is a minor configuration change. A conversion tool imports existing job definitions and calendars and reports on what it found, which replaces manual rebuilding for estates of any real size. The exception is GROW with SAP, public cloud, where the scheduling surface differs and the job-definition approach shifts accordingly.

Next steps

To see how a converted estate runs across SAP and non-SAP systems in practice, explore [Control-M for SAP](#).